COMPUTER VISION AND IMAGE UNDERSTANDING
Vol. 69, No. 2, February, pp. 202–221, 1998
ARTICLE NO. IV970558

Parameterizing Arbitrary Shapes via Fourier Descriptors for Evidence-Gathering Extraction

Alberto S. Aguado,* Mark S. Nixon, and M. Eugenia Montiel*

Electronics and Computer Science, University of Southampton, Highfield, Southampton SO17 1BJ, United Kingdom

Received February 7, 1996; accepted September 24, 1996

According to the formulation of the Hough Transform, it is possible to extract any shape that can be represented by an analytic equation with a number of free parameters. Nevertheless, the extraction of arbitrary shapes has centered on nonanalytic representations based on a table which specifies the position of edge points relative to a fixed reference point. In this paper we develop a novel approach for arbitrary shape extraction which combines the analytic representation of shapes with the generality of the characterization by Fourier descriptors. The formulation is based on a definition of the Hough Transform obtained by considering the parametric representation of shapes and extends the descriptional power of the Hough Transform beyond simple shapes, thus avoiding the use of tables. Since we use an analytic representation of shapes, the developed technique inherits the robustness of the original formulation of the Hough Transform. Based on the developed formulation, and by using different strategies of parameter space decomposition, various methods of shape extraction are presented. In these methods the parameter space is reduced by using gradient direction information as well as the positions of grouped edge points. Different methods represent a compromise between speed, noise sensitivity, simplicity, and generality. Some examples of the extraction process on a selection of synthetic and real images are presented, showing the successful extraction of target shapes from noisy data. © 1998 Academic Press

Key Words: **Computer vision; Hough Transform; Generalized Hough Transform; model based recognition; arbitrary shape extraction; shape detection; object recognition; shape representation; boundary functions; Fourier descriptors.**

1. INTRODUCTION

An important problem in image understanding is to extract or recognize global structures from local image data. In a model-based paradigm [1], descriptions of objects are characterized based on previous knowledge about their features. A common way of defining these features is by means of geometric primitives which represent the boundaries of shapes [2]. Geometric primitives can be described by analytic curves with a set of free parameters which define all the possible instances of a shape that might appear in an image. Accordingly, objects are recognized by determining the value of the free parameters. This process is equivalent to finding the optimum value of a cost function that represents the fit of a primitive to image data [2]. An effective way of computing the value of the free parameters is based on gathering evidence derived according to the mapping defined between the locus of a curve and the value of its parameters. This mapping constitutes the underlying concept of the Hough Transform (HT) .

In the HT, a parameter space is defined as a multidimensional space where each dimension represents a free parameter of a primitive. The parameters of a particular shape are computed by a common intersection of a set of loci in the parameter space. Each locus is defined for each image edge point and it comprises all the points in the parameter space which define a primitive that passes through the edge point. Only one point in each locus defines the shape in the image and this must be the same point for all loci. The HT computes robustly the intersection of all the loci as a maximum in an array which represents a discrete version of the parameter space wherein the loci are accumulated. The technique provides adequate results even in noisy conditions or where there are gaps in the boundary due to occlusion.

The HT was originally formulated to detect lines. It was then extended to include more complex primitives such as circles and ellipses [3–5]. Since these primitives are characterized by an increasing number of free parameters, and this number has an exponential relationship with the size of the parameter space, most of the research in the HT focuses on reducing its computational requirements. Other research has considered different implementation details such as optimizing the search for local maxima, and the study of the bounding and resolution of the accumulator array which represents the parameter space. An extensive review of the literature on the HT can be found in [6, 7].

In this paper we focus on the extension of the HT to extract arbitrary shapes. In a model-based approach to shape identification it is necessary to recognize large scale structures which define complex forms. Thus, the extension of the HT extraction process to arbitrary shapes has an important significance. This extension has been based on two main approaches. First, it is possible to constrain the HT technique to a feature detection method based on simple geometric primitives such as lines and

* Present address: INRIA Rhone-Alpes, ZIRST, 655 Avenue de l'Europe, 38330 Montbonnot Saint Martin, France.

1077-3142/98 $25.00

quadratic shapes. In this approach complex forms are defined following a structural feature decomposition characterized by these simple shapes [2, 8–10]. This type of decomposition is recognized as necessary for an efficient representation of the HT [7] and it relies on using additional postprocessing to extract complete models. In a second approach, an arbitrary model shape is defined by a nonanalytic representation in a form of a table. This model shape extends the simple definition of analytic curves, allowing entire shapes to be extracted. The original idea of this extension was presented in [11]. The Generalized Hough Transform (GHT) [12] improved the technique by constraining the parameter computation by including the orientation of edge points in the discrete representation. In this approach, when orientation and scale invariance are required, the extraction process involves a four-dimensional parameter space. Invariance to scale and orientation is achieved by including straightforward transformations of the tabular representation. Recent research has focused on reducing the computational burden of the technique [13–19].

Both approaches have important features. Structural decomposition maintains a complete specification of the parts through an analytic form. Nevertheless, in this specification primitives may not be sufficiently complex to provide an adequate description of the whole model. The importance of a complex characterization of shapes has been previously discussed [20]. On the other hand, the nonanalytic representation extends the description power via a complex characterization of shapes. This avoids additional postprocessing, but implies a partial knowledge of the border points (limited by the discretization and the size of the table). Additionally, the discrete nature of the nonanalytic representation may cause distortion in the accumulation process due to scale and rotation transformations.

In this paper we present a novel approach that extends the analytic form of simple geometric primitives to arbitrary shapes. This is achieved by reformulating the concepts of the extraction of nonanalytic shapes in terms of an analytic representation defined by the Fourier expansion of a curve. This analytic representation provides a description where models can be sufficiently complex for adequate shape characterization. In addition to developing the mapping for the accumulation process for a direct formulation of the HT, we will consider other formulations based on constraining the parameter space by exploiting edge direction information and the position of a collection of points.

Considering the formal definition of the HT [21], we introduce a mathematical formalism which embraces the methods for extracting nonanalytic shapes. This allows us to generalize the HT to extract analytically defined arbitrary shapes. In theory, the original formulation of the HT can be used to extract any primitive that can be represented by an analytic equation with a number of free parameters. Some research [4, 22] has suggested that the HT can be generalized to extract arbitrary analytic shapes by changing the equation of the curve under detection. In fact, the evolution of the HT from lines to ellipses corresponds to this generalization. Nevertheless, this concept has not been developed further, and the extraction of arbitrary shapes has preferred the use of a nonanalytic representation or a simple structural decomposition. This might have been motivated by the impression that the technique will present an exponential growth in memory and computation requirements when applied to more complex shapes. Intuitively, since the analytic extension from lines to ellipses implies a large increase in memory space and computational effort, the analytic generalization to extract arbitrary shapes appears to be infeasible. As such, the only viable alternative to extract shapes without performing a decomposition into simple geometric primitives is to follow a nonanalytic formulation. Actually, the number of free parameters required in the HT is independent of the complexity of a shape. The free parameters are related to the transformations which define all the instances of a shape that can appear in an image. Here it will be shown that if the allowed transformations are restricted to similarity transformations, then the extraction process requires a four-dimensional accumulator space.

Our approach to detect analytically defined primitives reformulates the concepts developed for the extraction of nonanalytic shapes by using a parametric representation based on Fourier descriptors. The analysis of curves by Fourier theory has been used in image understanding for several years [23–33]. The main interest in this analysis has focused on computing features or descriptors which provide a useful characterization for shape discrimination. These descriptors are defined by the amplitude and the phase of the harmonics obtained by expanding the parametric representation of a curve in Fourier series. Fourier descriptors can be generated by different parametric representations of curves. These representations have been defined via two approaches. The first approach is to transform the two-dimensional information of a curve into a one-dimensional periodic function, which is expanded in a Fourier series [24]. The second approach considers that the curve is described in the complex plane. Then, a Fourier expansion is performed in a complex-valued function [23]. The coefficients obtained with the second characterization are denoted as the elliptic Fourier descriptors of a curve [29].

Here, we shall use elliptic descriptors to characterize arbitrary shapes for primitive extraction. These descriptors were used in a parametric deformable model [34] to detect objects which are poorly represented in terms of fixed shapes. Probabilistic constraints derived from sample images were used to perform a local search in the parameter space. Although a local search is a plausible alternative in a model-free interpretation, the parameter space potentially has many local maxima [2]. This makes an evidence gathering technique, which performs a robust global search, more attractive when primitives are confined to rigid models.

The use of Fourier descriptors as a parameterized curve in the HT has been previously considered. The model presented in [35] highlights the bounding properties of the parameter space defined by Fourier descriptors; however, the underconstrained nature of the approach leads to an ill-posed technique with an enormous parameter space. This technique appears to await future application to images. Here, we show that it is possible to develop evidence gathering techniques for arbitrary shapes in a model-based approach based on a representation of curves characterized by elliptic Fourier descriptors. In order to obtain an analytic mapping between the image space and the parameter space, we relate the function which defines the position of points in a curve in a Fourier description to the definition of a nonanalytic curve. This mapping replaces the table used in a discrete representation by a continuous function. This function provides a compact and precise description of the border of a shape which minimizes errors caused by a discretization. The formulation includes variations in shape (e.g., scale and rotation) as changes in parameters of a continuous function, avoiding inherent discretization errors which are produced when transformations are performed on a tabular description.

This paper is organized as follows. In Section 2 we extend the formal definition of the HT to include the concepts of the extraction of arbitrary shapes. In Section 3 we introduce the parameterization of shapes by elliptic Fourier descriptors. A set of free parameters are obtained by applying similitude transformations on arbitrary shapes. By following the formal definition of the extraction of nonanalytic shapes presented in Section 2, we obtain an analytic formulation of the HT for arbitrary primitives. Section 4 is devoted to achieving a reduction in the parameter locus by exploiting the information provided by gradient direction and the position of several image points. In Section 5 we discuss the implementation and we present the results for the new formulation applied to synthetic and real images. Finally, Section 6 includes conclusions and further work.

2. ARBITRARY SHAPE EXTRACTION VIA THE HOUGH TRANSFORM

2.1. Hough Transform for Nonanalytic Shapes

In this section we extend the formal definition of the HT presented in [21] to encompass the concepts of the extraction of arbitrary nonanalytic shapes. Instead of representing shapes by using an implicit form of a curve, here we characterize the extraction process by a parametric form. Thus, a primitive Z is defined as the set of points in a continuous curve $\bar{z}(s, \bar{a})$ such that each point is identified by a parameter s. That is,

$$Z = \{\bar{z}(s, \bar{a}) \mid s \in D_s\}, \tag{1}$$

where D_s is the domain of the function. For clarity, two types of parameters are used as arguments of the function $\bar{z}(s, \bar{a})$: The parameter s specifies different points on the curve, and the parameter vector $\bar{a}$ defines the shape of the curve.

According to the definition in Eq. (1) the problem of primitive extraction is to obtain the parameter vector $\bar{a}$ from a set of points Z. In the HT, the extraction process is achieved by means of the relationship between the definition of a primitive and the parameter space. This relationship is illustrated in Fig. 1. For a point in Z given when s takes a particular value s_0 (i.e., $\bar{z}(s_0, \bar{a})$), a locus A_{s_0} is defined in the parameter space. This locus is composed of all the points which represent the parameters that define a primitive which passes through the point $\bar{z}(s_0, \bar{a})$.

If the function $\bar{z}(s, \bar{a})$ is invertible, the locus A_{s_0} can be defined as

$$A_{s_0} = \{\bar{z}^{-1}(s, \bar{z}(s_0, \bar{a})) \mid s \in D_s\}, \quad s_0 \in D_s. \tag{2}$$

The inverse function defines the kernel of the transformation and is specified by the curve under detection. In the HT, the locus A_{s_0} is defined for a set of edge points in an image I. If these points are represented by a primitive defined by a parametric function, then $I = \{\bar{\lambda}(t) \mid t \in D_I\}$ and the locus A_{s_0} can be redefined as

$$A_t = \{\bar{z}^{-1}(s, \bar{\lambda}(t)) \mid s \in D_s\}, \quad t \in D_I. \tag{3}$$

One point in A_t defines the primitive Z, and this point must be the same for all the loci obtained for any value of t. Consequently, the vector $\bar{a}$ in Eq. (1) corresponds to the intersection of the loci A_t defined for all the points in I. That is,

$$\bar{a} = \bigcap_{t \in D_I} A_t. \tag{4}$$

This intersection is computed in the HT by counting the number of times that the traces of the loci A_t pass through the cells of an array defined congruent to the parameter space. The array will contain a peak where different loci A_t intercept. The problem of primitive extraction is then transformed into the problem of searching for local peaks in an array. The robustness of the technique is based on the fact that a peak will appear if the image I contains points which do not belong to the primitive Z (i.e., noise) or if it contains only a subset of all the points (i.e., occlusion).

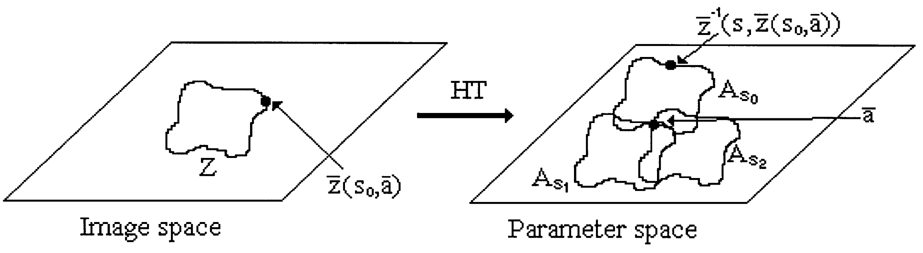

FIG. 1. Definition of curves in the image space and parameter space.

The accumulation process in the HT can be formalized by using a matching function which provides a value to be accumulated for each point in A_t. That is, for a point $\bar{b}$ in the parameter space and a point $\bar{d}$ in the set A_t,

$$D(\bar{b}, \bar{d}) = \begin{cases} 1 & \text{if } \bar{b} = \bar{d} \\ 0 & \text{if } \bar{b} \neq \bar{d} \end{cases}. \tag{5}$$

Thus, based on this function it is possible to determine whether a point $\bar{b}$ should be incremented according to a locus A_t. That is, the HT of the function $\bar{\lambda}(t)$ can be defined as

$$S_{\mathrm{HT}}(\bar{b}) = \int\!\!\int D(\bar{b}, \bar{z}^{-1}(s, \bar{\lambda}(t)))\, ds\, dt, \tag{6}$$

and $\bar{a}$ can be computed by finding a local maximum in this function. In implementation, this expression is discretized by the image and the accumulator array. Thus, the integrals are replaced by summations and the values of $\bar{\lambda}(t)$ are given by a set of discrete edge points $\bar{\lambda}_t$. In this case, the possible points $\bar{b}$ correspond to the cells of the array which represent the parameter space. A discrete version of the HT can be written as

$$S_{\mathrm{DHT}}(\bar{b}) = \sum_{t \in D_I} \sum_{s \in D_s} D(\bar{b}, \bar{z}^{-1}(s, \bar{\lambda}_t)), \tag{7}$$

where the summation in s defines a set of points congruent to the discretization of the parameter space.

2.2. Nonanalytic Representation of Shapes

2.2.1. Hough Transform for nonanalytic shapes. The use of the HT based on an analytic representation of primitives has been developed for lines and quadratic forms. Within this representation, the extraction of arbitrary shapes is constrained to a structural feature method where objects are decomposed into parts based on simple shapes. In general, a better representation of objects can be obtained by more complex shapes which comprise more elaborate components. These components might correspond to arbitrary shapes, and extend the descriptional power of the extraction process by taking advantage of all the information of a shape. The extraction of arbitrary shapes by the HT, without decomposition into simple primitives, has conventionally been based on nonanalytic representations [11–19]. This type of representation was introduced by Merlin and Farber [11]. In their method, a model shape is described by a discrete set of points $W = \{\bar{w}_B | B \in D_B\}$, whose positions are defined relative to an arbitrary reference point $\bar{R}$. Thus, the primitive in Eq. (1) can be redefined as a set of points obtained by translating the set W to $\bar{R}$. That is,

$$Z = \{\bar{z}_B \mid \bar{z}_B = \bar{R} + \bar{w}_B, B \in D_B\}. \tag{8}$$

This definition of a shape can be considered in the gathering process of the HT by identifying the point $\bar{R}$ with the parameter $\bar{a}$ in Eq. (1). Thus, the kernel in Eq. (3) is defined for analytic shapes by

$$R_t = \{\bar{\lambda}_t - \bar{w}_B \mid B \in D_B\}, \quad t \in D_I. \tag{9}$$

This expression corresponds to a transformation of the set of points W composed of a translation to the point $\bar{\lambda}_t$ and a reflection about the origin. In the Merlin–Farber method the reflection is defined as a rotation by π radians. Thus, the extraction process is performed by accumulating the traces of a rotation of the representation W translated to each point $\bar{\lambda}_t$.

By defining Eq. (9) as the kernel of the HT, it is possible to rewrite Eq. (7) for a nonanalytic representation as

$$S_{\mathrm{NA}}(\bar{b}) = \sum_{t \in D_I} \sum_{B \in D_B} D(\bar{b}, \bar{\lambda}_t - \bar{w}_B). \tag{10}$$

By comparing Eqs. (10) and (7) it can be seen that in order to gather evidence in the same way as in the original formulation of the HT, it is necessary to provide a shape representation obtained by the sampling defined by the discretization in the parameter space. That is, the points in W must correspond to the discretization of the function $\bar{z}^{-1}(s, \bar{\lambda}_t)$. Since the parameter space and the image space are congruent, the method requires that the discrete representation corresponds exactly to a discretization of the shape in the image.

2.2.2. Generalized Hough Transform. Ballard [12] improved the Merlin–Farber method by including gradient direction in the nonanalytic representation of a primitive. In the GHT, the locus in the parameter space defined by each point in an image is reduced to a set of plausible values constrained by edge direction information. That is, in Eq. (10) only a subset of points which form W are considered in the definition of $S_{\mathrm{NA}}(\bar{b})$. These points are determined by comparing the gradient direction of a primitive and the gradient direction computed on the point in an image. This makes the extraction process faster and increases robustness by reducing the probability of obtaining false instances of a shape due to coincidental arrangements of points [12].

The GHT can be formulated by considering the derivative at the points defined by the nonanalytic representation in Eq. (8). If the gradient direction z'_B in a point $\bar{z}_B$ is obtained by a function φ, then $z'_B = \varphi(\bar{z}_B)$. Since the points in Eq. (9) must be constrained to those points where the gradient direction in $\bar{\lambda}_t$, denoted as λ'_t, is equal to the derivative in $\bar{z}_B$, then $\lambda'_t = \varphi(\bar{z}_B)$. By considering the value of $\bar{R}$ as constant, the relationship between the gradient direction of a point in an image and the definition in Eq. (8) is given by

$$\lambda'_t = \varphi(\bar{z}_B) = \varphi(\bar{w}_B). \tag{11}$$

By using this expression it is possible to redefine the locus in Eq. (9) as the set of points

$$R_t = \{\bar{\lambda}_t - \varphi^{-1}(\lambda'_t)\}, \quad t \in D_I, \tag{12}$$

which defines a version of Eq. (3) for the GHT. Notice that in this equation the inverse function $\varphi^{-1}(\lambda'_t)$ can provide several points of the representation W. That is, several points in W might have the same slope λ'_t. By considering Eq. (12), the accumulation process defined by the GHT is given by

$$S_{\mathrm{GHT}}(\bar{b}) = \sum_{t \in D_I} D(\bar{b}, \bar{\lambda}_t - \varphi^{-1}(\lambda'_t)). \tag{13}$$

This equation contains only one summation because the locus of the parameter space is reduced. Namely, the kernel does not define all the points of the representation, but only those obtained by the inverse function $\varphi^{-1}(\lambda'_t)$. In Ballard's method, this function is stored in a table (R-table) which is indexed by the gradient direction computed at a point in an image. That is, given the gradient direction computed at a point, a set W_t is determined. This set contains all the entries that have the same gradient direction. Thus, $W_t = \{\bar{w}_B \mid \varphi(\bar{w}_B) = \lambda'_t, B \in D_B\}$. The points in W_t are mapped into the parameter space by the relationship $R_t = \{\bar{\lambda}_t - \bar{w}_B \mid \bar{w}_B \in W_t\}$. That is, an alternative version of Eq. (13) can be written as,

$$S_{\mathrm{GHT}}(\bar{b}) = \sum_{t \in D_I} \sum_{w_B \in W_t} D(\bar{b}, \bar{\lambda}_t - \bar{w}_B). \tag{14}$$

The accumulation processes defined in Eqs. (10) and (13) show that the algorithms for gathering evidence based on nonanalytic representations correspond to special implementations of the HT. The accumulation defined in Eq. (10) will be identical to the HT when W is sampled, according to the discretization in the image. In the case of Eq. (13) the same requirement implies that the values λ'_t must have at least one associated value in W through the inverse function $\varphi^{-1}(\lambda'_t)$ (i.e., $\forall \bar{\lambda}_t, W_t \neq \{\varnothing\}$).

For primitives which do not suffer any transformation (e.g., without change of scale and rotation) their discrete definition can be based on a table obtained from a template shape sampled by the quantization in the image. Nevertheless, it is difficult to obtain an adequate representation for a transformed shape based on this definition. A straightforward redefinition of the table for transformed shapes can cause difficulties in the accumulation process. In the case of the accumulation process in Eq. (10), two points can merge into one cell, or some cells can be missed. This problem of distortion is illustrated in Fig. 2. Figure 2a shows the nonanalytic representation of a shape as a set of points in an image space. Figure 2b shows the points in the discrete parameter space obtained by the application of the mapping defined in Eq. (9). The points in Figs. 2c and 2d were obtained by a straightforward scale transformation of the discrete representation and show the type of problems arising due to the discrete representation. In the expanded region in Fig. 2c, it can be seen that two points appear in one discrete cell, while in the expanded region in Fig. 2d not all the cells (through which the curve passes) are defined by the discrete representation.

In the case of the accumulation process in Eq. (13), when a primitive suffers transformations, and due to the limitations of the computation of gradient direction by local operators, the gradient direction values may not necessarily correspond exactly to the values in the discrete representation. That is, for a point in a shape, different gradient directions can be computed when the shape is transformed and discretized. Thus, although the nonanalytic formulations for primitive extraction correspond to particular developments of the HT, which is itself optimum in terms of detection error [4], the discrete nature of the approach reduces robustness with respect to the original formulation. As a consequence, these techniques can suffer when the image contains noise and shapes are occluded [36, 37].

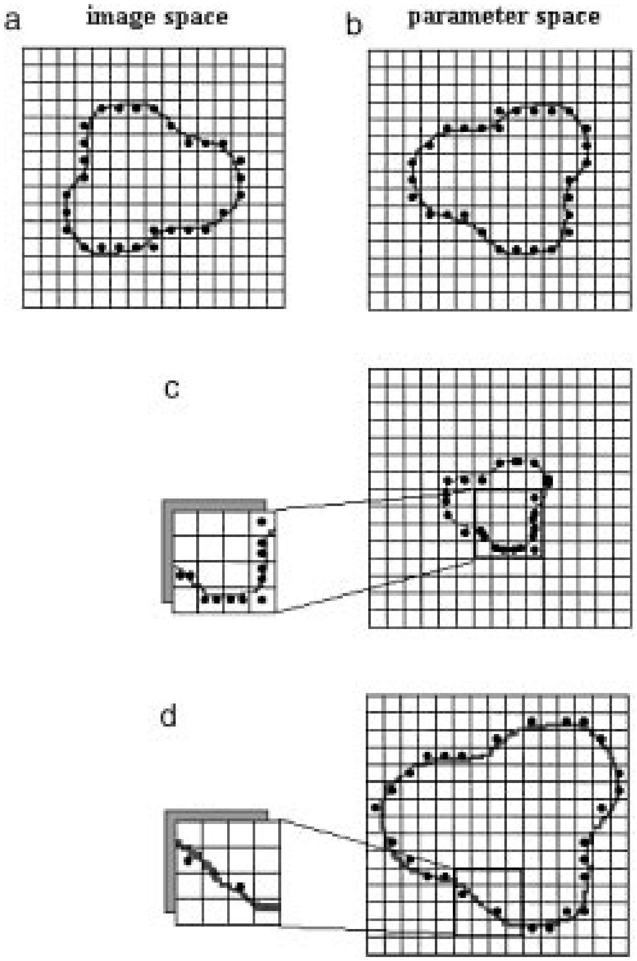

FIG. 2. Example of a distortion due to straightforward transformations on a nonanalytic representation.

3. ANALYTIC REPRESENTATION OF SHAPES VIA FOURIER DESCRIPTORS

3.1. *Fourier Parameterization of Arbitrary Shapes*

The first step for performing a generalization of the HT based on an analytic representation is to obtain an equation of a curve which can represent arbitrary model shapes. In general, there exist many ways of defining a curve. In this paper the extension is developed by using a parametric representation based on Fourier descriptors. This type of representation provides an accurate and compact characterization of curves, and can be defined by a vectorial parameterization which is adequate for the HT formulation developed in the previous section. Desirable properties such as convergence and generality have been widely recognized in their application to shape discrimination. An interesting computational feature is the direct access to frequencies, which facilitates the description of curves at multiple scales [33]. This feature is important because it gives the possibility of later use of multiresolution approaches of the HT [38, 39]. The main motivations for using Fourier descriptors are the generality of their definition, the simplicity in the calculation of derivatives, and the vectorial parameterization which can be directly related to nonanalytic representations.

The most direct and flexible representation of an entire shape is a parametric form of a curve. This representation can be mathematically described by a vector function which defines the position of the points in a curve by their components in two orthonormal axes. That is,

$$\bar{c}(s) = c_x(s)\,U_x + c_y(s)\,U_y, \tag{15}$$

where $U_x = [1, 0]$, $U_y = [0, 1]$ are two orthonormal vectors and the values of the parameter s serve to distinguish different points on the curve.

In order to obtain a representation of arbitrary shapes it is possible to parameterize the curve $\bar{c}(s)$ by the expansion of the components $c_x(s)$ and $c_y(s)$ in Fourier series. Then, the curve in Eq. (15) will be represented by its Fourier expansion given by the function

$$\bar{v}(s, \bar{v}) = v_x(s, \bar{v}_x)\,U_x + v_y(s, \bar{v}_y)\,U_y, \tag{16}$$

where $\bar{v}$ is a parameter vector which characterizes the form of the curve and is defined by the elliptic Fourier descriptors [23, 28, 29]. This vector is decomposed into two subsets of parameters: The vectors $\bar{v}_x$ and $\bar{v}_y$ which define the components of the function along the x and y axes, respectively (i.e., $\bar{v} = \bar{v}_x \cup \bar{v}_y$).

For $s \in [0, 2\pi)$, the Fourier expansion of $c_x(s)$ and $c_y(s)$, $\bar{v}_x$ and $\bar{v}_y$, can be expressed in trigonometric form as

$$\begin{aligned} v_x(s, \bar{v}_x) &= \sum_{k=1}^{n} a_{xk}\cos(ks) + b_{xk}\sin(ks), \\ v_y(s, \bar{v}_y) &= \sum_{k=1}^{n} a_{yk}\cos(ks) + b_{yk}\sin(ks), \end{aligned} \tag{17}$$

for $\bar{v}_x = (a_{x1}, b_{x1}, \ldots, a_{xn}, b_{xn})$, $\bar{v}_y = (a_{y1}, b_{y1}, \ldots, a_{yn}, b_{yn})$, and n is the maximum frequency in $\bar{c}(s)$. The DC terms have been omitted since the curve is defined with its center on the origin of the coordinate system (i.e., without translation). Each term in the summations in Eqs. (17) describes an elliptic phasor. The elliptic locus is defined by four parameters $(a_{xk}, a_{yk}, b_{xk}, b_{yk})$ and rotates at a frequency determined by k. An example of a shape generated by Eq. (17) is shown in Fig. 3a. In this figure, a point A in a closed curve is the result of a vectorial summation of three elliptic phasors (i.e., p_1, p_2, and p_3). Each rotating phasor defines an ellipse, and each ellipse is characterized by a different value of k. Although this Fourier expansion was originally devised for representing closed contours, the extension to open curves can be achieved easily. Two possible representations of open curves are illustrated in Figs. 3b and 3c. In Fig. 3b the entire curve is defined but the start and end points restrict t to a set of values which trace a particular segment. In the example in the figure, when $s \in [0, \frac{\pi}{2})$ one quarter of the curve is obtained. In Fig. 3c the open curve is composed of a double trace in the opposite direction. In this case, the curve must be parameterized such that the trace of the arc in one direction is defined for $s \in [0, \pi)$, and the curve is retraced in the opposite direction for $s \in [\pi, 2\pi)$.

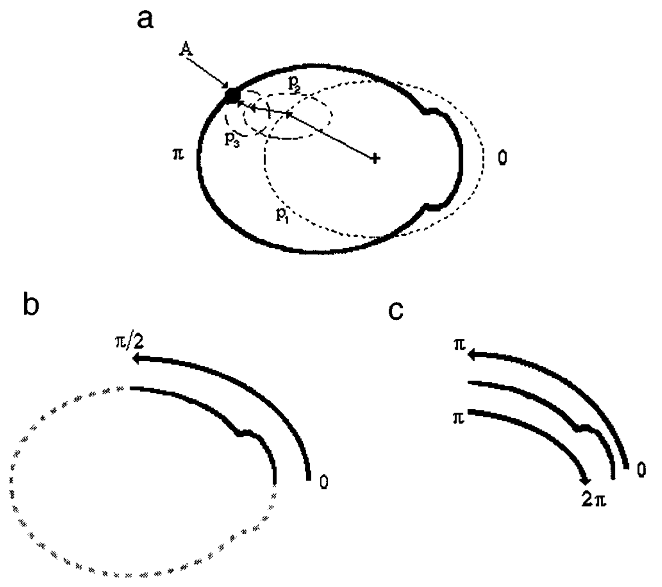

FIG. 3. Example of a contour defined by elliptic Fourier descriptors. (a) Closed contour. (b) Open curve defined by a subset of points. (c) Open curve defined by a double trace.

The elliptic Fourier descriptors of the curve in Eq. (15) are defined by

$$\begin{aligned} a_{xk} &= \frac{1}{\pi}\int_{-\pi}^{\pi} c_x(s)\cos(ks)\,ds, \\ a_{yk} &= \frac{1}{\pi}\int_{-\pi}^{\pi} c_y(s)\cos(ks)\,ds, \\ b_{xk} &= \frac{1}{\pi}\int_{-\pi}^{\pi} c_x(s)\sin(ks)\,ds, \\ b_{yk} &= \frac{1}{\pi}\int_{-\pi}^{\pi} c_y(s)\sin(ks)\,ds. \end{aligned} \tag{18}$$

Generally, the computation of these values must be performed for a model shape stored in a discrete image. If a model is composed of m points, the integrals can be approximated by summations. That is,

$$
\begin{aligned}
a_{xk} &= \frac{2}{m}\sum_{s=1}^{m} x_s \cos\left(k_s \frac{2\pi}{m}\right),\\
a_{yk} &= \frac{2}{m}\sum_{s=1}^{m} y_s \cos\left(k_s \frac{2\pi}{m}\right),\\
b_{xk} &= \frac{2}{m}\sum_{s=1}^{m} x_s \sin\left(k_s \frac{2\pi}{m}\right),\\
b_{yk} &= \frac{2}{m}\sum_{s=1}^{m} y_s \sin\left(k_s \frac{2\pi}{m}\right),
\end{aligned} \tag{19}
$$

where (x_s, y_s) represents the position of a point in a model shape. Notice that according to the Nyquist sampling theorem the possible values of k are the integers between 1 and $m/2$. That is, the value of n in Eq. (17) is given by $m/2$ and $k \in \{1, 2, \ldots, m/2\}$. A particular form of the computation in Eq. (19) can be based on the incremental change defined by a set of points organized in a chain-coded contour [29].

Since the problem of primitive extraction focuses on recognizing a class of shapes, the curve in Eq. (16) must be parameterized to form an appropriate shape model. That is, the Fourier coefficients will be used to describe the essential features of a family of primitives which will be parameterized by a transformation that generates all the instances of objects that might appear in an image. Each member in the family will be characterized by particular values of the parameters of the transformation. In the extraction of rigid parts, instances of shapes are formed by the different appearance of an original shape which has been rotated and scaled. These changes correspond to similarity transformations. A curve parameterized by similarity transformations requires four parameters: Two parameters define its position; one parameter specifies the scale; and the fourth parameter describes the rotation.

The parameterization of the curve in Eq. (16) by a similarity transformation can be obtained by defining two orthogonal functions $\zeta_x(s, \bar{a}_x)$ and $\zeta_y(s, \bar{a}_y)$ according to the matrix expression

$$
\begin{aligned}
&[\zeta_x(s, \bar{a}_x) \quad \zeta_y(s, \bar{a}_y)]\\
&= [a_0 \quad b_0] + l[v_x(s, \bar{v}_x) \quad v_y(s, \bar{v}_y)] \begin{bmatrix} \cos(\rho) & \sin(\rho) \\ -\sin(\rho) & \cos(\rho) \end{bmatrix},
\end{aligned} \tag{20}
$$

where $\bar{a}_x = (a_0, l, \rho)$, $\bar{a}_y = (b_0, l, \rho)$, the point (a_0, b_0) describes a translation, the value of l is a scale factor, and ρ specifies the rotation. Hence,

$$
\begin{aligned}
\zeta_x(s, \bar{a}_x) &= a_0 + l(v_x(s, \bar{v}_x)\cos(\rho) - v_y(s, \bar{v}_y)\sin(\rho)),\\
\zeta_y(s, \bar{a}_y) &= b_0 + l(v_x(s, \bar{v}_x)\sin(\rho) + v_y(s, \bar{v}_y)\cos(\rho)).
\end{aligned} \tag{21}
$$

In this definition, the Fourier coefficients in the vectors $\bar{v}_x$ and $\bar{v}_y$ which characterize the form of the curve in Eq. (16) are not considered as parameters of $\zeta_x(t, \bar{a}_x)$ and $\zeta_y(t, \bar{a}_y)$ since they are constant for all the curves which constitute the family of primitives. Namely, the curves are parameterized only by the parameters in the transformation. Thus, by taking this parameterization, the extraction problem can be defined based on a geometric primitive obtained by the orthogonal composition of the functions in Eq. (21). That is,

$$
\bar{z}(s, \bar{a}) = \zeta_x(s, \bar{a}_x)\, U_x + \zeta_y(s, \bar{a}_y)\, U_y, \tag{22}
$$

for $\bar{a} = (a_0, b_0, l, \rho)$. The extraction process is performed by determining the value of the parameters $\bar{a}$ in this equation that best match the model $\bar{v}(s, \bar{v})$ in Eq. (16) to image data.

3.2. Hough Transform for Shapes Represented by Fourier Descriptors

The extraction of a model shape characterized by Fourier descriptors can be formulated based on the gathering evidence process defined by the HT. In this section this formulation is developed by extending the concepts of the extraction of arbitrary nonanalytic shapes presented in Section 2 to shapes represented by analytic curves. In order to define the HT for arbitrary analytic shapes it is necessary to establish the kernel of the transformation in Eq. (3). This kernel can be obtained by a particular form of the parameterization in Eq. (22) which defines a curve without the translation term. This curve will be denoted by

$$
\bar{w}(s, l, \rho) = \xi_x(s, l, \rho)\, U_x + \xi_y(s, l, \rho)\, U_y, \tag{23}
$$

where the functions $\xi_x(s, l, \rho)$ and $\xi_y(s, l, \rho)$ are given by

$$
\begin{aligned}
\xi_x(s, l, \rho) &= lg(s, \rho),\\
\xi_y(s, l, \rho) &= lh(s, \rho),
\end{aligned} \tag{24}
$$

for $g(s, \rho)$ and $h(s, \rho)$ which are two functions that define the rotation around each coordinate axis. That is,

$$
\begin{aligned}
g(s, \rho) &= v_x(s, \bar{v}_x)\cos(\rho) - v_y(s, \bar{v}_y)\sin(\rho),\\
h(s, \rho) &= v_x(s, \bar{v}_x)\sin(\rho) + v_y(s, \bar{v}_y)\cos(\rho).
\end{aligned} \tag{25}
$$

Thus, the definition of $\bar{z}(s, \bar{a})$ in Eq. (22) can be rewritten in terms of $\bar{w}(s, l, \rho)$ in Eq. (23) as

$$
\bar{z}(s, \bar{a}) = (a_0, b_0) + \bar{w}(s, l, \rho). \tag{26}
$$

Since this equation corresponds to the function which defines the set Z in Eq. (8) with the reference point given by the values of (a_0, b_0), then the kernel and the accumulation process in the HT can be obtained analogously to Eqs. (9) and (10), respectively. Nevertheless, in this case the locus in the parameter space is defined by an analytic representation. According to Eq. (3), the locus in the parameter space for an analytic representation, when the points in the image are defined by $I = \{\bar{\lambda}(t) \mid t \in D_I\}$, is given by

$$A_t = \{\bar{\lambda}_t - \bar{w}(s, l, \rho) \mid s \in D_s\}, \quad t \in D_I. \tag{27}$$

This kernel is based on a continuous function $\bar{w}(s, l, \rho)$ whose parameters l and ρ define a curve in specific orientation and scale. Then, if each possible instance of the function is considered, the HT for the image I can be defined by

$$S_F(\bar{b}, l, \rho) = \int\int D(\bar{b}, \bar{\lambda}(t) - \bar{w}(s, l, \rho))\, dt\, ds, \tag{28}$$

for $\bar{b} = (a_0, b_0)$. This expression combines the analytic form of Eq. (6) with the kernel of an arbitrary shape defined in the accumulation process in Eq. (10). The discrete version of this formulation is analogous to Eq. (7). That is,

$$S_{DF}(\bar{b}, l, \rho) = \sum_{t \in D_I} \sum_{s \in D_s} D(\bar{b}, \bar{\lambda}_t - \bar{w}(s, l, \rho)). \tag{29}$$

Thus, given an image point $\bar{\lambda}_t$ the accumulation process traces a curve $\bar{\lambda}_t - \bar{w}(s, l, \rho)$ in the plane (a_0, b_0) in the four-dimensional array defined by (a_0, b_0, l, ρ) . The parametric nature of $\bar{w}(s, l, \rho)$ provides an adequate representation for the fast and accurate computation of the trace of the curve in the plane [40]. Nevertheless, the dimensionality of the parameter space imposes an important constraint which implies that unless the number of parameters is reduced or the quantization of the parameter space is coarse, it will be necessary to consider an implementation that includes other HT developments. For example, extensions based on a parallel implementation [7] could provide accurate results in a reduced processing time. An alternative way of reducing the computational requirements is to decompose the parameter space by the inclusion of gradient direction information or the use groups of edge points.

4. PARAMETER SPACE REDUCTION

4.1. Inclusion of Gradient Direction Information

By following the development in Section 2 it is possible to consider the use of gradient direction information to constrain the locus of the parameter space defined in Eq. (29). Gradient direction can be included in the formulation of the extraction of arbitrary analytic curves by rewriting Eq. (29) in terms of the mapping φ^{-1} in Eq. (12). In this case the mapping must be defined according to the Fourier representation of a curve.

The gradient direction at a point $\bar{w}(s, l, \rho)$ will be denoted as $w'(s, \rho)$. That is, $w'(s, \rho) = \varphi(\bar{w}(s, l, \rho))$. This value can be obtained as the tangent of the angle defined by the tangent vector function $\bar{w}'(s, l, \rho) = \xi_x'(s, l, \rho)\, U_x + \xi_y'(s, l, \rho) U_y$. According to Eq. (24) the components of this vectorial expression are given by

$$\begin{aligned} \xi_x'(s, l, \rho) &= \frac{\partial \xi_x(s, l, \rho)}{\partial s} = l g'(s, \rho), \\ \xi_y'(s, l, \rho) &= \frac{\partial \xi_y(s, l, \rho)}{\partial s} = l h'(s, \rho), \end{aligned} \tag{30}$$

and the tangent angle is

$$w'(s, \rho) = \tan^{-1}\left(\frac{h'(s, \rho)}{g'(s, \rho)}\right), \tag{31}$$

where

$$\begin{aligned} g'(s, \rho) &= (v_x'(s, \bar{v}_x)\cos(\rho) - v_y'(s, \bar{v}_y)\sin(\rho)), \\ h'(s, \rho) &= (v_x'(s, \bar{v}_x)\sin(\rho) + v_y'(s, \bar{v}_y)\cos(\rho)), \end{aligned} \tag{32}$$

and

$$\begin{aligned} v_x'(s, \bar{v}_x) &= \sum_{k=1}^{n} k(-a_{xk}\sin(ks) + b_{xk}\cos(ks)), \\ v_y'(s, \bar{v}_y) &= \sum_{k=1}^{n} k(-a_{yk}\sin(ks) + b_{yk}\cos(ks)). \end{aligned} \tag{33}$$

According to Eq. (11) the gradient direction λ_t' computed at an image point must be equal to the gradient direction at the point $\bar{z}(s, \bar{a})$. By considering the definition in Eq. (26), the value of λ_t' can be related to $\bar{w}(s, l, \rho)$ by $\lambda_t' = \varphi(\bar{z}(s, \bar{a})) = \varphi(\bar{w}(s, l, \rho))$. That is,

$$\lambda_t' = w'(s, \rho) = \tan^{-1}\left(\frac{v_x'(s, \bar{v}_x)\sin(\rho) + v_y'(s, \bar{v}_y)\cos(\rho)}{v_x'(s, \bar{v}_x)\cos(\rho) - v_y'(s, \bar{v}_y)\sin(\rho)}\right). \tag{34}$$

In order to obtain the inverse mapping $\bar{w}(s, l, \rho) = \varphi^{-1}(\lambda_t')$, this equation is expressed in terms of a function γ which depends on s,

$$\lambda_t' = \gamma(s) + \rho, \tag{35}$$

for

$$\gamma(s) = \tan^{-1}\left(\frac{v_y'(s, \bar{v}_y)}{v_x'(s, \bar{v}_x)}\right). \tag{36}$$

Then, based on a value of λ_t' it is possible to obtain a set of values of s given by

$$s = \gamma^{-1}(\lambda_t' - \rho). \tag{37}$$

In general, this mapping does not have a closed-form solution and it must be solved using a numerical method. According to the definition of $\gamma(s)$ in Eq. (36), the solution of Eq. (37)

corresponds to the zeroes of the equation

$$\tan(\lambda'_t - \rho)v'_x(s, \bar{v}_x) - v'_y(s, \bar{v}_y) = 0. \tag{38}$$

Since this equation only has one variable (s) an adequate estimate of the solutions can be easily determined. Notice that Eq. (38) is independent of scale, consequently it only has to be computed for each value of ρ in the accumulation process.

By considering Eq. (38), the inverse function in Eq. (37) can be expressed as

$$\begin{aligned}&\gamma^{-1}(\lambda'_t - \rho)\\ &\quad = \{s \mid \tan(\lambda'_t - \rho)v'_x(s, \bar{v}_x) - v'_y(s, \bar{v}_y) = 0, s \in D_s\}.\end{aligned} \tag{39}$$

Thus, the function $\varphi^{-1}(\lambda'_t)$ is obtained by substitution of the values given by this equation in $\bar{w}(s, l, \rho)$. That is, the points in the model with edge direction λ'_t are given by

$$\varphi^{-1}(\lambda'_t) = \bar{w}(\gamma^{-1}(\lambda'_t - \rho), l, \rho). \tag{40}$$

Accordingly, the locus in Eq. (27) constrained by edge direction is redefined by

$$A_t = \{\bar{\lambda}_t - \bar{w}(\gamma^{-1}(\lambda'_t - \rho), l, \rho)\}, \quad t \in D_I, \tag{41}$$

and the HT in Eq. (29) becomes

$$S_{\mathrm{DFG}}(\bar{b}, l, \rho) = \sum_{t \in D_I} D(\bar{b}, \bar{\lambda}_t - \bar{w}(\gamma^{-1}(\lambda'_t - \rho), l, \rho)). \tag{42}$$

In this expression, the inclusion of gradient directional information reduces the number of points to be accumulated in the parameter space: Instead of considering all the points defined by $\bar{w}(s, l, \rho)$, only some points $\bar{w}(\gamma^{-1}(\lambda'_t - \rho), l, \rho)$ are used. This decreases noise in the accumulator array. Naturally, this requires the computation of directional information which can itself be susceptible to noise.

In general, the inclusion of gradient direction information in analytic formulations of the HT has been used to reduce the dimensions of the parameter space (e.g., [5, 41–43]). In the previous formulation, although the parameter space is incremented by using more selected information, the accumulation process is still in a four-dimensional space. Nevertheless, the space can be reduced since the parameter s has been eliminated. Then, any other parameter can be used to draw a curve in a three-dimensional parameter space. This idea can be explained by considering the definition of $\bar{w}(s, l, \rho)$ in the angle–magnitude form. The vector defined by $\bar{w}(s, l, \rho)$ can be expressed in an angle–magnitude form (ψ, r) according to Eq. (23) as

$$\begin{aligned}\psi(s, \rho) &= \tan^{-1}\left(\frac{h(s, \rho)}{g(s, \rho)}\right),\\ r(s, l, \rho) &= l\sqrt{g^2(s, \rho) + h^2(s, \rho)}.\end{aligned} \tag{43}$$

That is,

$$\begin{aligned}\psi(s, \rho) &= \tan^{-1}\left(\frac{v_x(s, \bar{v}_x)\cos(\rho) - v_y(s, \bar{v}_y)\sin(\rho)}{v_x(s, \bar{v}_x)\sin(\rho) + v_y(s, \bar{v}_y)\cos(\rho)}\right),\\ r(s, l, \rho) &= l\sqrt{\begin{aligned}&(v_x(s, \bar{v}_x)\cos(\rho) - v_y(s, \bar{v}_y)\sin(\rho))^2\\ &+ (v_x(s, \bar{v}_x)\sin(\rho) + v_y(s, \bar{v}_y)\cos(\rho))^2.\end{aligned}}\end{aligned} \tag{44}$$

Analogous to Eq. (35) we can express the angle $\psi(s, \rho)$ by using a function which depends on s. That is,

$$\psi(s, \rho) = \Psi(s) - \rho, \tag{45}$$

for

$$\Psi(s) = \tan^{-1}\left(\frac{v_x(s, \bar{v}_x)}{v_y(s, \bar{v}_y)}\right). \tag{46}$$

Based on this function and on the definition of the parameter s in Eq. (37), the angle $\psi(s, \rho)$ can be reparameterized to be independent of s by the composite mapping

$$\psi(\lambda'_t, \rho) = \Psi(\gamma^{-1}(\lambda'_t - \rho)) - \rho. \tag{47}$$

This mapping obtains the tangent of the angle of a point in the model from the derivative computed at a point. Figure 4 illustrates this mapping. Figures 4b and 4c represent the functions Ψ and γ for each point in the curve shown in Fig. 4a. Figure 4d represents the composite mapping and displays the result of the parametric function $(\gamma(s), \Psi(s))$. According to Eq. (47), a value λ'_t can be mapped into Ψ by first finding a value of s in the function γ^{-1} in Fig. 4c and then using this as the parameter in the function Ψ in Fig. 4b. It can be seen that for a particular value of λ'_t (e.g., 1.25) there exist several possible values of s. As an example, the mapping for $s = 4$ is mapped (into Fig. 4b) to obtain the value $\psi(s, \rho) = 1.75$. This is the same result as obtained by the composite mapping in Fig. 4d.

By considering Eq. (26) we can relate the value of $\psi(s, \rho)$ and the locus in the parameter space by

$$\frac{y_t - b_0}{x_t - a_0} = \frac{h(s, \rho)}{g(s, \rho)} = \tan(\psi(\lambda'_t, \rho)) \tag{48}$$

for $\bar{\lambda}_t = (x_t, y_t)$.

Then, by substitution of the mapping in Eq. (45), the locus in the parameter space given by Eq. (48) can be redefined by a line. That is,

$$\begin{aligned}A_t = \{\bar{b} \mid y_t - b_0 - (x_t - a_0)\tan(\Psi(\gamma^{-1}(\lambda'_t - \rho)) - \rho) = 0\},&\\ t \in D_I.&\end{aligned} \tag{49}$$

Based on this locus it is possible to gather evidence of the three parameters a_0, b_0, and ρ; in general (except for circles

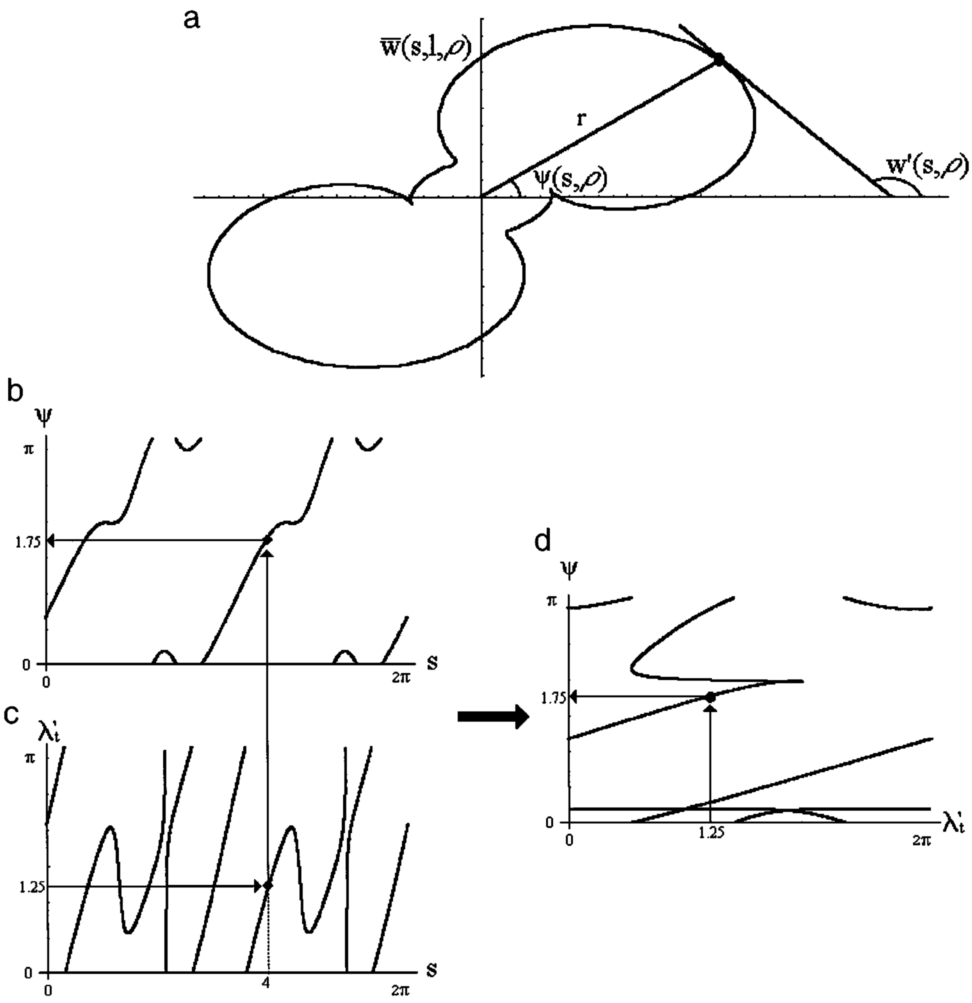

FIG. 4. Example of the composite map $\Psi(\gamma^{-1}(\lambda'_t))$. (a) Example shape. (b) Function of the angle for the model in (a). (c) Function of the derivative for the model in (a). (d) Composition of the mapping shown in (b) and (c).

and ellipses) one value of λ'_t will define several lines. By reparameterizing the line in Eq. (49) by a parameter $\tau = x_t - a_0$, an alternative form of the HT in Eq. (42) can be defined by

$$S_{\mathrm{DFG}}(\bar{b},\rho) = \sum_{t\in D_I}\sum_{\tau} D(\bar{b}, \bar{\lambda}_t - (x_t - \tau, y_t - \tau\tan(\Psi(\gamma^{-1}(\lambda'_t - \rho)) - \rho))), \qquad (50)$$

which is equivalent to gathering the evidence given by the trace of the lines in Eq. (49) for all $t \in D_I$.

The value of the parameter l can be obtained by performing a second stage in the accumulation process. Once the values of the parameters a_0, b_0, and ρ of an instance of a shape in an image are known, the parameter l can be solved by using the magnitude expression in Eq. (44). That is,

$$l = \frac{\sqrt{(x_t - a_0)^2 + (y_t - b_0)^2}}{\sqrt{(v_x(s,\bar{v}_x)\cos(\rho) - v_y(s,\bar{v}_y)\sin(\rho))^2 + (v_x(s,\bar{v}_x)\sin(\rho) + v_y(s,\bar{v}_y)\cos(\rho))^2}}. \qquad (51)$$

In order to evaluate the denominator of this equation, the values of $v_x(s, \bar{v}_x)$ and $v_y(s, \bar{v}_y)$ can be computed by taking the values of s defined according to the tangent of the angle in Eq. (48). That is,

$$\frac{y_t - b_0}{x_t - a_0} = \frac{v_x(s,\bar{v}_x)\sin(\rho) + v_y(s,\bar{v}_y)\cos(\rho)}{v_x(s,\bar{v}_x)\cos(\rho) - v_y(s,\bar{v}_y)\sin(\rho)}, \qquad (52)$$

and the value of s corresponds to the zeroes of the function

$$\frac{y_t - b_0}{x_t - a_0} = \tan\left(\rho + \tan^{-1}\left(\frac{v_y(s,\bar{v}_y)}{v_x(s,\bar{v}_x)}\right)\right), \qquad (53)$$

which can be expressed as

$$\alpha(\bar{\lambda}_t,\rho) = \left\{ s \,\middle|\, y_t - b_0 - (x_t - a_0)\tan\left(\rho + \tan^{-1}\left(\frac{v_y(s,\bar{v}_y)}{v_x(s,\bar{v}_x)}\right)\right)\right\}. \qquad (54)$$

Thus, the position of a point $\bar{\lambda}_t$ in the image is used to obtain the value of s which defines $v_x(s, \bar{v}_x)$ and $v_y(s, \bar{v}_y)$. These values are used in Eq. (51) to gather evidence of the parameter l in a histogram (i.e., one-dimensional accumulator array). The maximum in this histogram corresponds to the value of l for the instance of a shape defined by a_0, b_0, and ρ. Accordingly, Eqs. (50) and (51) define an evidence gathering technique for arbitrary shape extraction which requires a three-dimensional array for the accumulation of the location and rotation parameters and a histogram for computing the scale.

4.2. *Use of Sets of Points*

In the previous section we have shown how the inclusion of gradient direction information in the formulation of the HT can be used to reduce the parameter space required in the extraction of arbitrary shapes. The four-dimensional parameter space defined for primitive extraction was reduced to a three-dimensional parameter space by defining an extra equation which determines the angular relationship between gradient information in an image and a model shape. This idea can be extended by using the equations provided by other information within an image. This section shows three alternative reductions in the parameter space that are obtained by combining the information provided by several edge points. Use of simultaneous points to constrain the parameter space has been a common approach in the extraction of analytic curves by the HT. In the case of simple analytic curves such as lines, circles, and ellipses, there exist geometrical relationships between groups of points and parameters. These relationships can be exploited to obtain analytic forms of HT mappings which reduce the dimensionality of the parameter space. In the case of arbitrary shapes, it is not possible to define constraints by geometric properties of shapes, and consequently, it is necessary to use a more general formulation.

Instead of using gradient direction information, the parameter space can be reduced by the constraint imposed by the position of a second point. This provides a pair of equations which can be solved to obtain the values of one parameter. By considering two image points $\bar{\lambda}_{t_1}$ and $\bar{\lambda}_{t_2}$, the locus in the parameter space defined in Eq. (27) is constrained to be the solution of the intersection $A_{t_1,t_2} = A_{t_1} \cap A_{t_2}$, for $A_{t_1} = \{\bar{\lambda}_{t_1} - \bar{w}(s_1, l, \rho) \,|\, s_1 \in D_s\}$, $t_1 \in D_I$, and $A_{t_2} = \{\bar{\lambda}_{t_2} - \bar{w}(s_2, l, \rho) \,|\, s_2 \in D_s\}$, $t_2 \in D_I$. That is, the locus in the parameter space is constrained by the solution of the system of two equations,

$$\begin{aligned}(a_0, b_0) &= \bar{\lambda}_{t_1} - \bar{w}(s_1, l, \rho),\\ (a_0, b_0) &= \bar{\lambda}_{t_2} - \bar{w}(s_2, l, \rho).\end{aligned} \tag{55}$$

Accordingly, the intersection (a_0, b_0) is defined by the vectorial equation

$$\bar{\lambda}_{t_1} - \bar{\lambda}_{t_2} - \bar{w}(s_1, l, \rho) + \bar{w}(s_2, l, \rho) = 0. \tag{56}$$

The solution of this equation can be obtained by considering the angular relationship between the points $\bar{\lambda}_{t_1}$ and $\bar{\lambda}_{t_2}$. If the components of these points are defined by $\bar{\lambda}_{t_1} = (x_{t_1}, y_{t_1})$ and $\bar{\lambda}_{t_2} = (x_{t_2}, y_{t_2})$, then the tangent of the angle with respect to the x-axis of the line which joins the two points is

$$\beta = \frac{y_{t_2} - y_{t_1}}{x_{t_2} - x_{t_1}}. \tag{57}$$

By taking the definitions of x_{t_1}, x_{t_2}, y_{t_1}, and y_{t_2} from Eq. (55) and the decomposition of $\bar{w}(s, l, \rho)$ in Eq. (23), this relationship becomes

$$\beta\xi_x(s_2, l, \rho) - \xi_y(s_2, l, \rho) - \beta\xi_x(s_1, l, \rho) + \xi_y(s_1, l, \rho) = 0. \tag{58}$$

Based on this equation it is possible to define a function $s_2 = q(s_1, \rho)$. By considering the definitions of $\xi_x(s, l, \rho)$ and $\xi_y(s, l, \rho)$ in Eq. (24) this function is given by

$$\begin{aligned} &q(s_1, \rho)\\ &\quad = \left\{ s_2 \left| \begin{array}{l} v_x(s_2, \bar{v}_x)(\beta\cos(\rho) - \sin(\rho))\\ - v_y(s_2, \bar{v}_y)(\beta\sin(\rho) + \cos(\rho)) + d(s_1, l, \rho) = 0 \end{array} \right. \right\}, \end{aligned} \tag{59}$$

where $d(s_1, l, \rho) = -\beta\xi_x(s_1, l, \rho) + \xi_x(s_1, l, \rho)$ is a constant term so the only variable is s_2 .

Based on this definition, it is possible to express the solution of the system in Eq. (55) independently of s_2. That is,

$$\begin{aligned} \vartheta(\bar{\lambda}_{t_1}, \bar{\lambda}_{t_2}, l, \rho) = \{ s_1 \,|\, &\bar{\lambda}_{t_1} - \bar{\lambda}_{t_2} - \bar{w}(s_1, l, \rho)\\ &+ \bar{w}(q(s_1, \rho), l, \rho) = 0, s_1 \in D_s \}. \end{aligned} \tag{60}$$

Thus, if we consider once more the components in Eq. (24), the equation in this function can be written as the system of equations given by

$$\begin{aligned} x_{t_1} - x_{t_2} - lg(s_1, \rho) + lg(q(s_1, \rho), \rho) = 0,\\ y_{t_1} - y_{t_2} - lh(s_1, \rho) + lh(q(s_1, \rho), \rho) = 0, \end{aligned} \tag{61}$$

and the function in Eq. (60) can be defined independent of l as

$$\begin{aligned} \vartheta(\bar{\lambda}_{t_1}, \bar{\lambda}_{t_2}, \rho) \left\{ s_1 \left| \frac{y_{t_1} - y_{t_2}}{h(s_1, \rho) - h(q(s_1, \rho), \rho)} \right. \right.\\ \left. - \frac{x_{t_1} - x_{t_2}}{g(s_1, \rho) - g(q(s_1, \rho), \rho)} = 0, s_1 \in D_s \right\}. \end{aligned} \tag{62}$$

Thus, the locus constrained by a pair of points is given by

$$\begin{aligned} A_{t_1,t_2} = \{ &\bar{\lambda}_{t_1} - \bar{w}(\vartheta(\bar{\lambda}_{t_1}, \bar{\lambda}_{t_2}, \rho),\\ &\Gamma(\bar{\lambda}_{t_1}, \bar{\lambda}_{t_2}, \rho, \vartheta(\bar{\lambda}_{t_1}, \bar{\lambda}_{t_2}, \rho)), \rho) \}, \quad t_1, t_2 \in D_I, \end{aligned} \tag{63}$$

for

$$\Gamma\left(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \rho, s\right) = \frac{y_{t_1} - y_{t_2}}{h(s, \rho) - h(q(s, \rho), \rho)}. \tag{64}$$

Accordingly, the HT is defined in a three-dimensional space as

$$S_{\mathrm{DF2P}}(\overline{b}, \rho) = \sum_{t_1, t_2 \in D_I} D\left(\overline{b}, \overline{\lambda}_{t_1} - \overline{w}\left(\vartheta(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \rho\right), \Gamma\left(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \rho, \vartheta\left(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \rho\right)\right), \rho\right)\right). \tag{65}$$

In this definition the value of s_1, obtained from $\vartheta(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \rho)$, defines a trace of a curve in the plane (a_0, b_0). This locus depends on ρ, and therefore it must be accumulated in a different plane (a_0, b_0) forming a three-dimensional parameter space (a_0, b_0, ρ). In a complete extraction process, a second accumulation stage, defined according to Eq. (51), can be used to obtain the parameter l. Thus, it is possible to reduce the parameter space either by using directional information or by taking pairs of points simultaneously. Both approaches constrain the parameter calculation by including an extra equation which provides more information about the geometry of a shape.

In a further extension we can consider another point to constrain the parameter space by three equations. The extra constraint replaces the curve traced in the plane (a_0, b_0) by a single point. This can be used to reduce noise in the accumulator array or to further reduce the dimensionality of the accumulation array by considering more selected information. By including a third restriction in Eq. (55) the locus in the parameter space is constrained to the intersection defined by $A_{t_1,t_2,t_3} = A_{t_1} \cap A_{t_2} \cap A_{t_3}$. Based on the intersection of A_{t_1} and A_{t_3}, it is possible to define an equation similar to Eq. (58) but the point $\overline{\lambda}_{t_2}$ is replaced by a third image point $\overline{\lambda}_{t_3} = (x_{t_3}, y_{t_3})$. This new equation provides a function equivalent to Eq. (59) which obtains the value of s_3 from the value of s_1. That is, $s_3 = p(s_1, \rho)$. Then, the system in Eq. (61) is constrained by two extra equations. That is,

$$\begin{aligned}
&x_{t_1} - x_{t_2} - lg(s_1, \rho) + lg(q(s_1, \rho), \rho) = 0,\\
&y_{t_1} - y_{t_2} - lh(s_1, \rho) + lh(q(s_1, \rho), \rho) = 0,\\
&x_{t_1} - x_{t_3} - lg(s_1, \rho) + lg(p(s_1, \rho), \rho) = 0,\\
&y_{t_1} - y_{t_3} - lh(s_1, \rho) + lh(p(s_1, \rho), \rho) = 0.
\end{aligned} \tag{66}$$

Accordingly, the values which define a point in the plane (a_0, b_0) can be obtained by redefining the function in Eq. (62) as

$$\vartheta(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \overline{\lambda}_{t_3} \rho) = \left\{ s_1 \left| \begin{array}{lcl} \dfrac{y_{t_1} - y_{t_2}}{h(s_1,\rho) - h(q(s_1,\rho),\rho)} - \dfrac{x_{t_1} - x_{t_2}}{g(s_1,\rho) - g(q(s_1,\rho),\rho)} & = & 0, \\ \dfrac{y_{t_1} - y_{t_3}}{h(s_1,\rho) - h(p(s_1,\rho),\rho)} - \dfrac{x_{t_1} - x_{t_3}}{g(s_1,\rho) - g(p(s_1,\rho),\rho)} & = & 0, s_1 \in D_s \end{array} \right. \right\}. \tag{67}$$

Based on this definition a locus similar to Eq. (63) and an accumulation process similar to Eq. (65) can be obtained. The only difference is that the new function will reduce the curve traced for each value ρ in the plane (a_0, b_0) into a single point. Namely, three points define the traces of two curves in the plane (a_0, b_0) whose points are distinguished by the value of ρ. The constraints in Eq. (66) specify the conditions that must be satisfied by the parameter s_1 in order to define an intersection of the two traces. Since the HT searches for the intersection of all the curves instead of gathering evidence of the traces of the two curves, only their intersection point can be accumulated.

In another alternative formulation we can consider the use of an extra point together with edge direction information. In this case the locus in the parameter space is constrained by the intersection of two loci defined by Eq. (49). That is, the locus in Eq. (49) is redefined by $A_{t_1,t_2} = A_{t_1} \cap A_{t_2}$ for

$$\begin{aligned}
A_{t_1} &= \left\{\overline{b} \mid y_{t_1} - b_0 - \tan\left(\Psi\left(\gamma^{-1}(\lambda'_{t_1} - \rho)\right) - \rho\right)\right.\\
&\qquad \left. \times \left(x_{t_1} - a_0\right) = 0\right\}, \quad t_1 \in D_I,\\
A_{t_2} &= \left\{\overline{b} \mid y_{t_2} - b_0 - \tan\left(\Psi\left(\gamma^{-1}(\lambda'_{t_2} - \rho)\right) - \rho\right)\right.\\
&\qquad \left. \times \left(x_{t_2} - a_0\right) = 0\right\}, \quad t_2 \in D_I.
\end{aligned} \tag{68}$$

The locus A_{t_1,t_2} is obtained by computing the intersection of the functions which define these loci. Thus, the HT defined by taking a pair of points and their gradient direction is given by

$$S_{\mathrm{DFG2P}}(\overline{b}, \rho) = \sum_{t_1, t_2 \in D_I} D\left(\overline{b}, \left(\sigma_x\left(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \lambda'_{t_1}, \lambda'_{t_2}, \rho\right), \sigma_y\left(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \lambda'_{t_1}, \lambda'_{t_2}, \rho\right)\right)\right), \tag{69}$$

for

$$\begin{aligned}
&\sigma_x\left(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \lambda'_{t_1}, \lambda'_{t_2}, \rho\right)\\
&= \frac{y_{t_2} - y_{t_1} + \Psi\left(\lambda'_{t_1}, \rho\right)x_{t_1} + \Psi\left(\lambda'_{t_2}, \rho\right)x_{t_2}}{\Psi\left(\lambda'_{t_1}, \rho\right) - \beta\Psi\left(\lambda'_{t_2}, \rho\right)},\\
&\sigma_y\left(\overline{\lambda}_{t_1}, \overline{\lambda}_{t_2}, \lambda'_{t_1}, \lambda'_{t_2}, \rho\right)\\
&= \frac{\Psi\left(\lambda'_{t_1}, \rho\right)y_{t_2} - \Psi\left(\lambda'_{t_2}, \rho\right)y_{t_1} - \Psi\left(\lambda'_{t_1}, \rho\right)\Psi\left(\lambda'_{t_2}, \rho\right)\left(x_{t_2} - x_{t_1}\right)}{\Psi\left(\lambda'_{t_1}, \rho\right) - \Psi\left(\lambda'_{t_2}, \rho\right)}.
\end{aligned} \tag{70}$$

The HT defined by two points and derivatives constrains the locus in the parameter space to a single point. Therefore, the resulting accumulator will present less noise than the three-dimensional parameter accumulator defined by Eq. (50) which traces a set of lines. Additionally, as in the case of the use of the information of three image points, the locus in the parameter space is independent of s and l, therefore, it can be used to trace a curve whose points are parameterized for the values ρ in a two-dimensional array.

5. IMPLEMENTATION AND RESULTS

In general, the four-dimensional parameter accumulator in Eq. (29) will impose a significant computational burden. This is impractical unless some parameters are known or the quantization of the parameter space is coarse. Hence, this section concentrates on the formulation which defines a three-dimensional accumulator array. In addition to showing how the technique can be implemented, this illustrates the effects of noise when point pairing strategies are considered.

As a first implementation (implementation A), we consider the formulation obtained by including directional information. The line in Eq. (49) provides a mapping which can be used to gather evidence of the center and the rotation parameters of a primitive. After edge pixels with directional information are obtained, each point is used to compute the values of $\psi(\lambda'_t, \rho)$ in Eq. (47). This is performed by evaluating the zeroes of the function $\gamma^{-1}(s)$, defined in Eq. (39), in the tangent angle definition $\Psi(s)$ in Eq. (46). Once the values of $\psi(\lambda'_t, \rho)$ are computed, the line in the parameter space given by the values of (a_0, b_0) in Eq. (49) is obtained. Since the value of $\psi(\lambda'_t, \rho)$ depends on the rotation ρ, each line defined by a particular value of ρ is used to increment the cells in the plane (a_0, b_0) in a three-dimensional array (a_0, b_0, ρ). After all the points are considered, local maxima in this accumulator correspond to the parameters of an instance of a shape. These parameters are used in a second accumulation stage to gather evidence of the scale parameter. In the second stage, each image point is used to find the zeroes of Eq. (53). These values are used in Eq. (51) to obtain the cell in the scale array which must be incremented. After all the image points have been considered, a local maximum in this array represents the scale parameter l of the primitive. In our implementation, the zeroes of Eqs. (38) and (53) were determined using a Newton–Raphson method. These equations correspond to a single variable real function expressed as a Fourier expansion so their derivatives can be easily obtained, analogous to Eq. (33).

In the extraction process, it is convenient to normalize the scale parameter l in such a way that a set of integer values can be related to the radius of the image shape in pixel units. The normalization can be achieved by redefining the scale parameter as $l/g(s_0, \rho)$ for a value of ρ of zero, and where s_0 addresses the point whose angle with respect to the x-axis is zero degrees. This point is illustrated in Fig. 5a. Within this definition, each integer value of l characterizes a shape whose line traced from the center of the shape to the point $\overline{w}(s_0, l, \rho)$ has a length of l pixels.

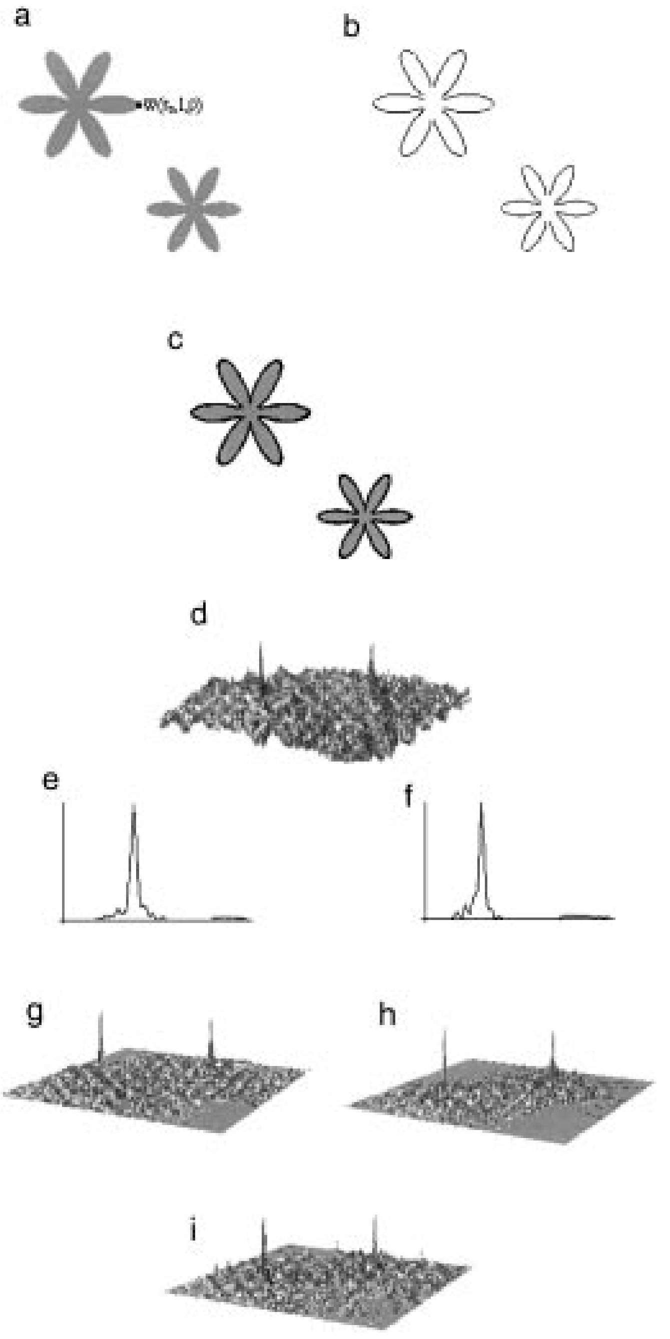

FIG. 5. Synthetic image example. (a) Original image. (b) Edge image. (c) Resulting image. (d) Plane of the accumulation array (implementation A). (e) Scale histogram for one maximum in (d). (f) Scale histogram for the second maximum in (d). (g) Accumulator results for implementation B (point pairing). (h) Accumulator results for implementation C (using three points). (i) Accumulator results for implementation D (gradient information and point pairing).

An example of the results of implementation A is illustrated in Fig. 5. Figure 5a shows a synthetic image (256 × 256 pixels) which contains two shapes defined by three Fourier coefficients. Figure 5b shows the edges computed by using the Canny edge

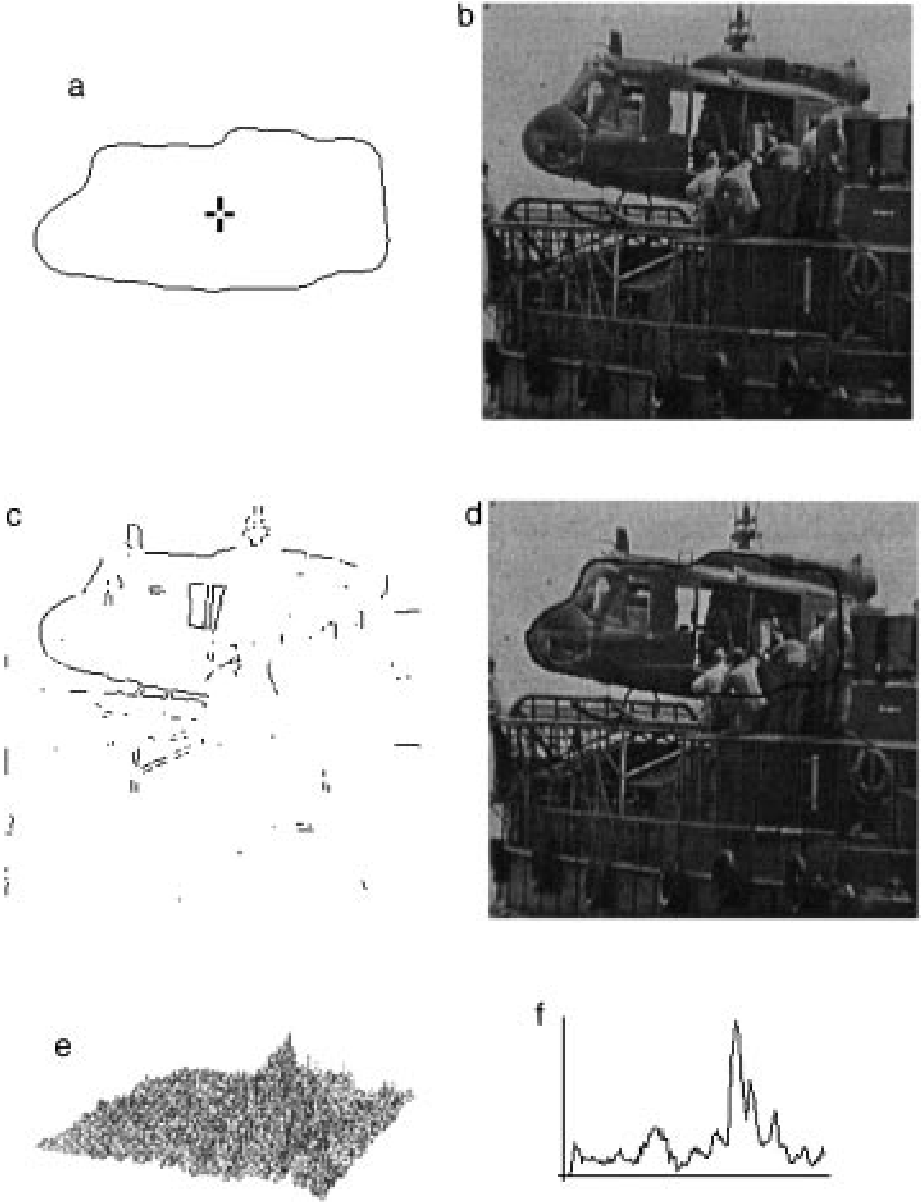

FIG. 6. Example of implementation A. (a) Model shape. (b) Original image. (c) Edge image. (d) Resulting image. (e) A plane of the accumulator array. (f) Scale histogram.

operator. The result of the extraction process is shown in Fig. 5c where the detected primitives are superimposed on black on the original image. Figure 5d shows the plane (a_0, b_0) of the three-dimensional accumulator space (a_0, b_0, ρ) where ρ is at its maximum value. The magnitude of each point represents the accumulated evidence for the position of the center of the shape. In this case the two peaks correspond to different instances of the shape. Figures 5e and 5f show the scale histograms obtained for the two maxima in Fig. 5d.

Figure 6 illustrates the application of the extraction process to a real image, the model shape corresponding to a helicopter cabin is shown in Fig. 6a. The model was extracted by manually tracing the edges in a high-contrast image and it was approximated by 15 harmonics in a Fourier expansion (whose amplitude was significantly greater than the remaining descriptors; an analysis of the number of harmonics needed in a Fourier approximation can be seen elsewhere [29]). In this figure the reference point is indicated by a cross. The definition of this model was used to accumulate evidence for the image shown in Fig. 6b. The edge information of this image is shown in Fig. 6c. The result is presented in Fig. 6d superimposed on the original image; Figs. 6e and 6f present the final accumulators from which the parameters were derived to draw the result. In this example, each accumulator contains a peak which defines the primitive accurately even when the image is corrupted by incomplete data and noise.

A second implementation (implementation B) can be based on the formulation of Eq. (65) which avoids using directional information by pairing points. Since edge gradient direction does not always provide accurate information, we should expect that avoiding its use will lead to a reduction in the noise in the accumulator array. Nevertheless, this reduction is not without computational cost. The computation of the locus in the parameter space by Eq. (63) is more complex than Eq. (49) because it involves the solution of the function in Eq. (62). An important consideration in this formulation is the number of pairs of points that should be evaluated in the accumulation process. In principle, for each point in the image it is only necessary to take another point to accumulate complete information (i.e., the number of pairs of points evaluated is equal to the number of points in an image). Nevertheless, it is necessary to ensure that each point in a pair belongs to the same primitive. Although this condition cannot be fully satisfied for complex images, the effects of pairing points which do not belong to the same primitive can be ameliorated by increasing the number of pairs per pixel. Additionally, it is convenient to establish a filtering constraint which verifies that pairs of points are close to each other. This restriction is based on the premise that close points are more likely to belong to the same primitive. In our implementation, each point in the image is paired to five other points which lie within a distance of 80 pixels. The effectiveness of the accumulation process for the synthetic image in Fig. 5a can be seen in the accumulator shown in Fig. 5g. The scale histograms are the same as Figs. 5e and 5f, and the extracted shape is equivalent to the one shown in Fig. 5c. Figure 5g shows that the use of pairs of points reduces the noise in the accumulator array while the peak remains in the same position. The application of this implementation to a real image is illustrated in Fig. 7. The model shape is defined by an open curve approximated by eight harmonics in a Fourier expansion, shown in Fig. 7a. In this example, the model shape (the wing) appears twice in the image, but with a different rotation. The located instances of the shape are superimposed on black on the original image in Fig. 7d. The accumulators in Figs. 7e and 7f correspond to the evidence for the instance of the shape in the superior part in the image while Figs. 7g and 7h correspond to the evidence for the instance of the shape in the inferior part. The accumulators in Figs. 7e and 7g correspond to two slices of the three-dimensional parameter space (a_0, b_0, ρ) for two different values of ρ.

A further reduction of the noise in the accumulator array can be achieved by including a third point in the formulation (implementation C). This constrains the parameter locus to the points given by the solution of the function in Eq. (67). In the implementation developed, the solution is found by considering the first constraint to obtain the value of s_1 analogous to Eq. (62) and then it is verified that this value satisfies the second constraint in Eq. (67). As shown in Fig. 5h, when a three-point restriction is included, the accumulator is less noisy than the accumulators in Figs. 5d and 5g. An example of the application on a real image is shown in Fig. 8. The model shape and description used in Fig. 8 is the same as that used in Fig. 7. There are again two instances of the shape in this image and both are found accurately, in spite of the image noise. In this case the accumulators for both instances present a clear peak.

The formulation which combines point pairing and gradient directional information in Eq. (69) (implementation D) can be used to reduce the noise in the accumulation process with less computational effort than implementations B and C, which use groups of points exclusively (pairs and triplets). The intersection in Eq. (70) can be used to accumulate more relevant information than that obtained by Eq. (50). Nevertheless, the accumulated points still depend on noisy gradient directional information and it is unlikely that the intersection in Eq. (70) gives an accurate position of the primitive, unless it is robustly computed. In order to ameliorate the effect of using gradient direction, it was necessary to increase the number of points per pixel considered in the accumulation process. In the implementation developed, the accumulation of the intersection of a line with any line defined by 25 other points provided adequate results. Figure 5i shows the result of implementation D for the image in Fig. 5a. The noise in the accumulator is slightly greater than that in Fig. 5g. Nevertheless, the computational time spent in the extraction process is less. The application of this formulation to a real image is shown in Fig. 9. The model shape used has the same description as the model shown in Fig. 9a. In this case, the instance in Fig. 9a suffered a three-dimensional rotation which caused only part of the model (the front of the cabin) to match the image data. Additionally, the borders in Fig. 9b present noise and some data is missed. Despite this, the location of the primitive is accurately obtained and the accumulators in Figs. 9d and 9e show a significant peak.

Besides the computational requirements and noise reduction, the consideration of noise sensitivity is important in applications. The results shown in Fig. 5 illustrate the effect of the inclusion of directional information in the accumulation process. In this sense, point pairing techniques seem to be sufficiently robust to eliminate the use of gradient information. Nevertheless, processing time and the requirement that the points in a subset must belong to the same primitive could make the use of gradient information more attractive than point pairing. This idea is illustrated in the example in Fig. 10, which shows the shape of the accumulator for the four implementations when a shape is extracted from noisy data. Figure 10a shows the type of noise used in the analysis. This image was generated by adding a random edge point for each edge point deleted from the primitive. In this image only 30% of the data pertains to the primitive. Figure 10b shows the same result obtained by each of the four implementations, superimposed on the edge data. The sensitivity of each implementation can be seen in the accumulators shown in Figs. 10c, 10d, 10e, and 10f. These results correspond to the

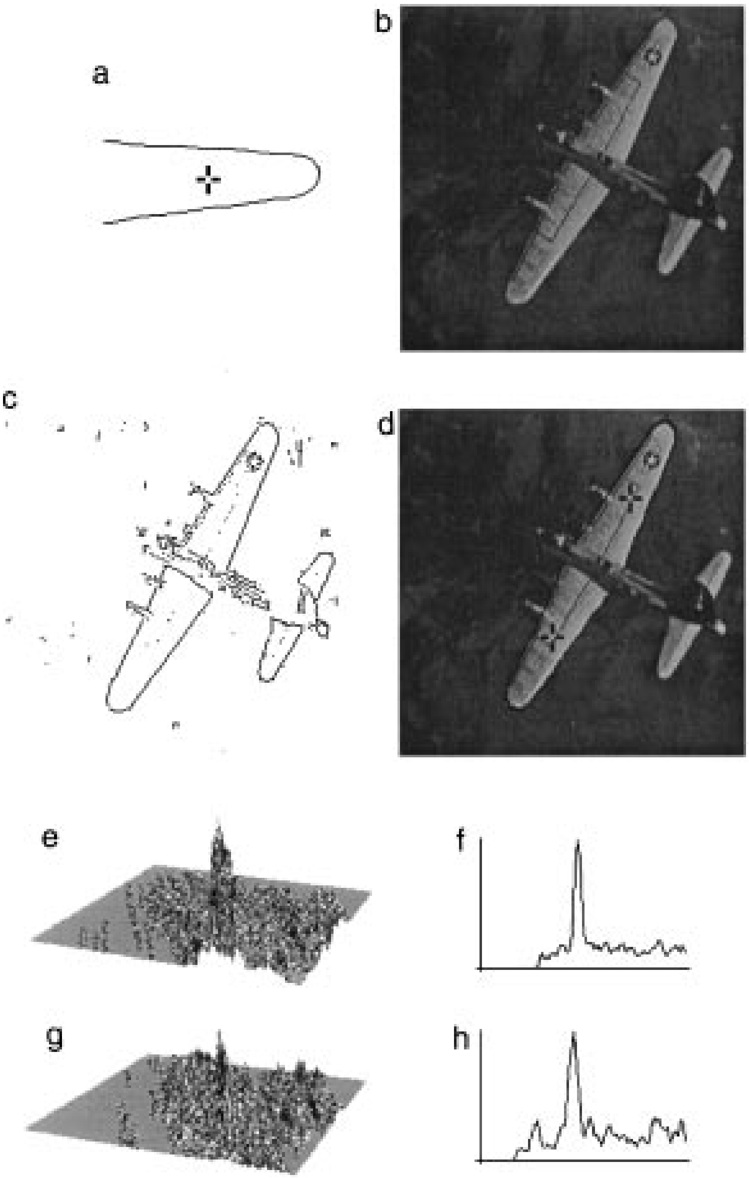

FIG. 7. Example of implementation B. (a) Model shape. (b) Original image. (c) Edge image. (d) Resulting image. (e) The plane of the accumulator array for one maximum. (f) Scale histogram for the maximum in (e). (g) The plane of the accumulator array for the second maximum. (h) Scale histogram for the maximum in (g).

projection of the two-dimensional location accumulator into a plane defined by the x-axis; the sequence shows the accumulators obtained by varying the noise from 0, 30, 50, 70, and 90%. In this figure it can be seen that although techniques which avoid using edge direction provide a better peak for less noise, they can present false peaks when noise is increased. Namely, the noise sensitivity in the implementation that uses point pairing decays faster than the implementation based on gradient direction information. As a consequence, false peaks can be more evident in the techniques which consider pairs of points. An example of this can be seen in Fig. 10d with a small peak developing to the left of the main peak with 70% of noise. It is important to notice that in this analysis the use of distance constraints for pairs and triplets of points was avoided, and only two sets are considered for each point in the image. In the case of pairs of points and gradient information, six points were paired per pixel. This consideration was made in order to show the results of the straightforward formulations. Nevertheless, the decrease in the sensitivity may be reduced by using filters that ensure that the paired points belong to the same primitive. The implementation of this may not be simple, and the effectiveness often depends on particular applications. Additionally, the results only show the noise sensitivity when there are points in the image which do not belong to the primitive, and they presuppose that the noise

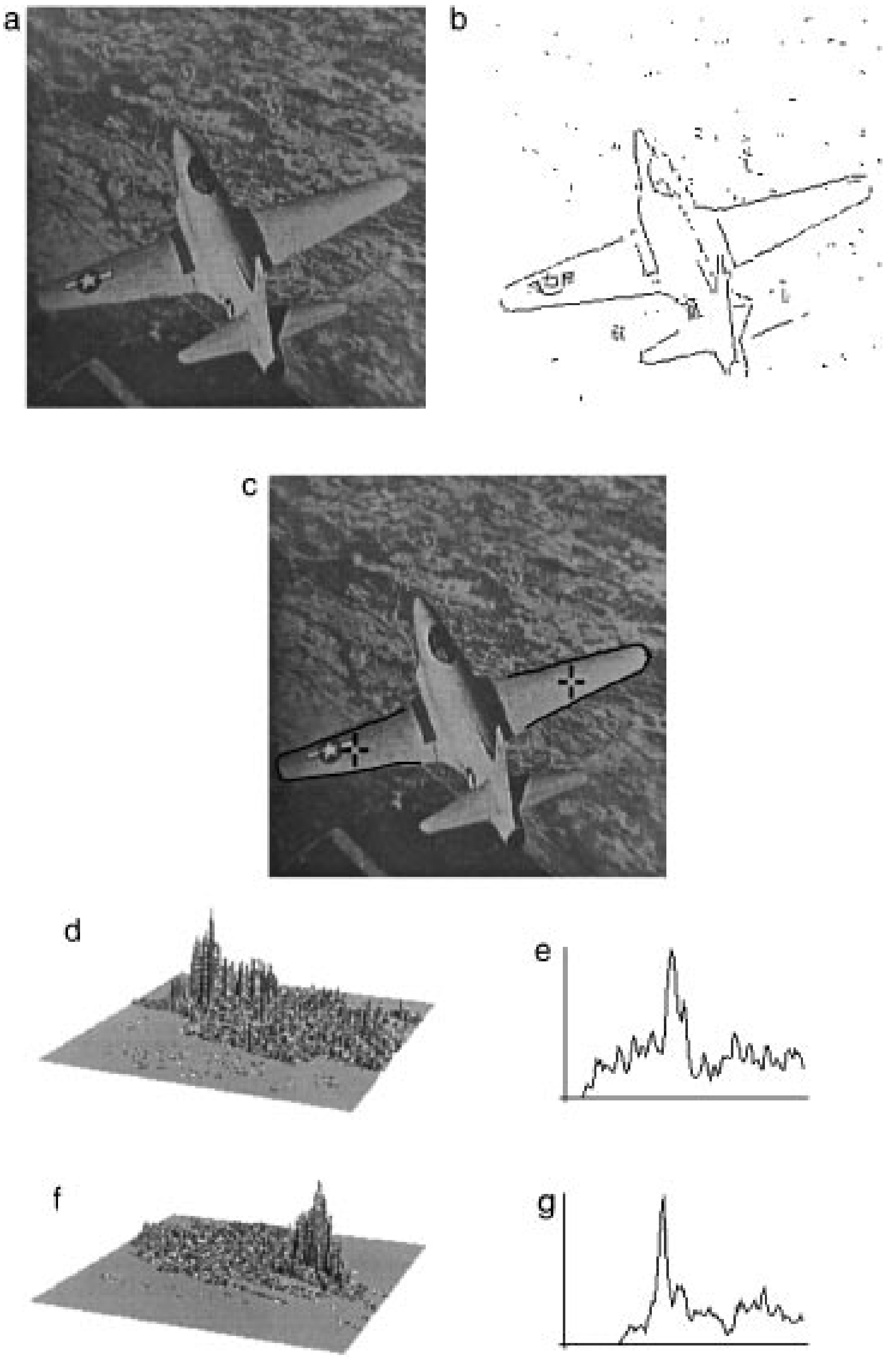

FIG. 8. Example of implementation C. (a) Original image. (b) Edge image. (c) Resulting image. (d) A plane of the accumulator array for the first maximum. (e) Scale histogram for the first maximum. (f) A plane of the accumulator array for the second maximum. (g) Scale histogram for the second maximum.

does not contribute to the error of the computation of gradient direction.

6. CONCLUSIONS AND FURTHER WORK

A novel method for extracting arbitrary shapes has been developed. This method corresponds to the formulation of the HT to model shapes characterized by elliptic Fourier descriptors and is based on the extension of the formal definition of the HT to embrace arbitrary shapes which are not analytically described. The technique extends the descriptional power of the analytic formulation of the HT beyond simple shapes, avoiding the use of tables. Since the formulation maintains an analytic representation, the method inherits the robustness of the

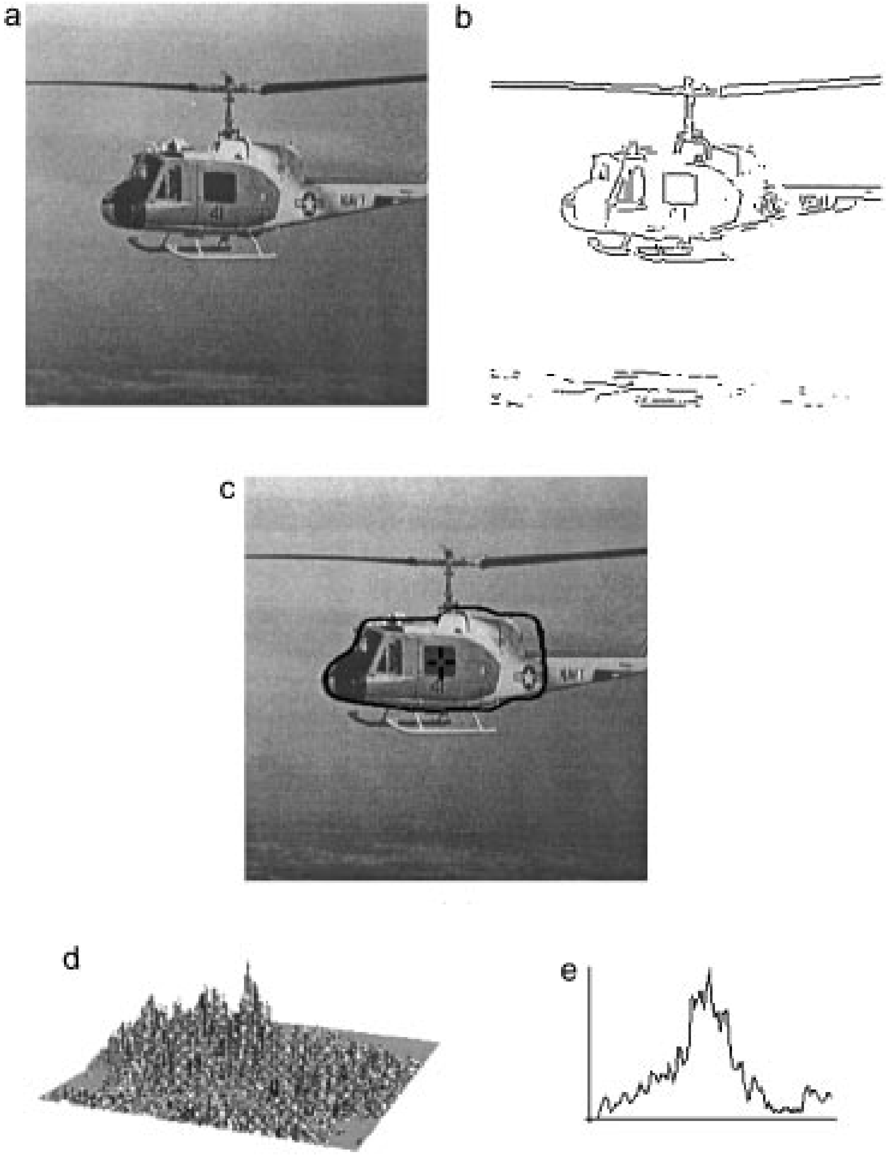

FIG. 9. Example of implementation D. (a) Original image. (b) Edge image. (c) Resulting image. (d) A plane of the accumulator array. (e) Scale histogram for the maximum.

original formulation of the HT and errors caused by a discretization are minimized.

The new technique can be used to extract shapes with differing orientation and size, in a four-dimensional parameter space. By using different strategies of parameter space reduction four methods of shape extraction have been developed. These methods include the information of the position of a set of points as well as gradient direction, and decompose the parameter space into a three-dimensional accumulator and a single histogram. The results demonstrate that the effectiveness of the accumulation process is maintained even when image data is corrupted by occlusion and noise. The choice between different implementations is a compromise: While the use of gradient direction allows the development of faster algorithms, the limitations of local operators can produce wide peaks in the resulting accumulators. Nevertheless, the peak's position is accurate and is maintained even with severe image clutter. When using a collection of points, the accumulator will present a narrow peak. Although this could be interpreted as an increase in robustness, the technique based on sets of points can suffer in noisy imagery and when points belonging to other primitives are presented. In general, when most data in an image define a primitive, the implementation obtained without using directional information appears better than when directional information is included. Nevertheless, when the image noise is increased, false peaks can emerge in the accumulator array, reducing the apparent improvement in performance. This effect can be reduced by incorporating some heuristic rules for selecting the points in a set as well as by increasing the number

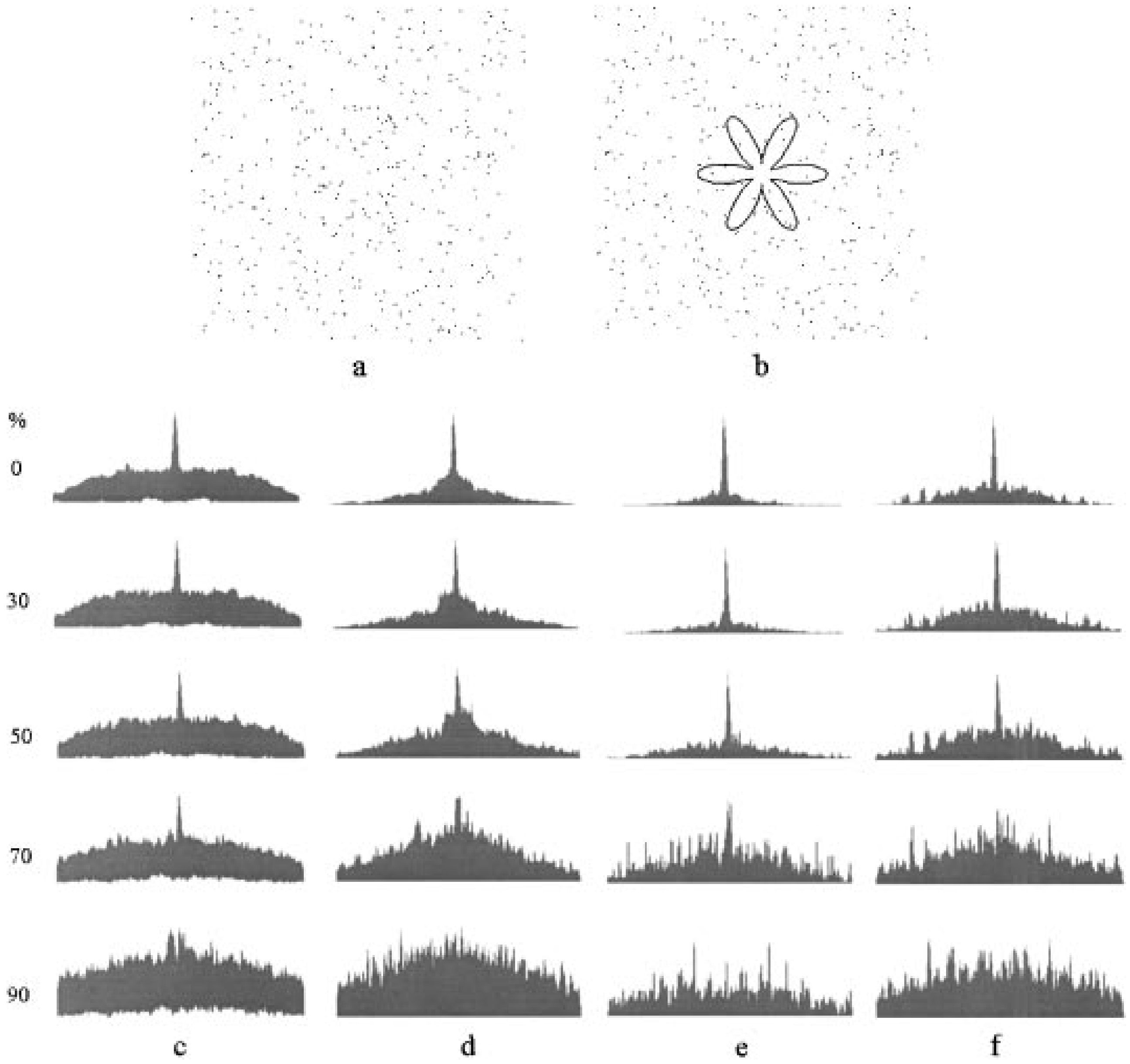

FIG. 10. Noise sensitivity. (a) Example of noisy image (70%). (b) Result. (c) Accumulators for implementation A. (d) Accumulators for implementation B. (e) Accumulators for implementation C. (f) Accumulators for implementation D.

of pairs of points formed per edge point, but with increase in computational cost. The type of algorithms which combine both the use of gradient direction and the information of the position of a small number of points provide a sensible alternative for general applications.

The implementations developed for extracting arbitrary shapes result in a three-dimensional space. This implies a complexity that might not meet the required speed for some applications, given current standard computational capabilities. Future work focuses on a further decrease of the computational load to make the technique more attractive. The basic idea is to consider in the formulation the concepts of the methods developed for reducing the computational load of the GHT. This can be achieved in a similar way, since the concepts of the GHT were considered in the presented formulation. Additionally, other approaches for computational reduction can be included in our extraction algorithm. We are particularly interested in the potential for extension to multiresolution descriptors. The capability of handling different levels of detail by manipulating the frequencies in the Fourier representation of a shape might be useful for shape extraction by using a multiresolution HT approach [38, 39]. Finally, other methods which extend the HT to three-dimensional object recognition [29, 44] might provide a suitable avenue for further research.

REFERENCES

1. R. T. Chin and C. R. Dyer, Model-based recognition in robot vision, *ACM Comput. Surveys* **18**, 1986, 67–108.
2. G. Roth and M. D. Levine, Extracting geometric primitives, *CVGIP: Image Understanding* **58**, 1993, 1–22.
3. R. O. Duda and P. E. Hart, Use of the Hough transform to detect lines and curves in pictures, *Commun. ACM* **15**, 1972, 11–15.
4. J. Sklansky, On the Hough technique for curve detection, *IEEE Trans. Comput.* **27**, 1978, 923–926.
5. S. Tsuji and F. Matsumoto, Detection of ellipses by a modified Hough transform, *IEEE Trans. Comput.* **27**, 1978, 777–781.
6. J. Illingworth and J. Kittler, A survey of the Hough transform, *Comput. Vision Graphics Image Process.* **48**, 1988, 87–116.
7. V. F. Leavers, Which Hough transform? *CVGIP: Image Understanding* **58**, 1993, 250–264.
8. C. Hoffman, *Geometric and Solid Modeling, an Introduction*, Morgan Kaufmann, San Mateo, CA, 1989.
9. D. Kriegman and J. Ponce, On recognizing and positioning curved 3-D objects from image contours, *IEEE Trans. Pattern Anal. Mach. Intell.* **12**, 1990, 1127–1137.
10. J. Ponce, A. Hoggs, and D. Driegman, On using CAD models to compute the pose of curved 3D objects, *CVGIP: Image Understanding* **55**, 1992, 184–197.
11. P. M. Merlin and D. J. Farber, A parallel mechanism for detecting curves in pictures, *IEEE Trans. Comput.* **24**, 1975, 96–98.
12. D. H. Ballard, Generalizing the Hough transform to detect arbitrary shapes, *Pattern Recognit.* **13**, 1981, 111–122.
13. L. S. Davis, Hierarchical generalized Hough transform to detect arbitrary shapes, *Pattern Recognit.* **15**, 1982, 277–285.
14. R. Kichnapuram and D. Casasent, Hough space transformations for discrimination and distortion estimation, *Comput. Vision Graphics Image Process.* **38**, 1987, 299–316.
15. M. G. Albanesi and M. Ferretti, A space saving approach to the Hough transform, in *Proceedings, 10th Int. Conf. on Patt. Recog., Atlantic City, NJ, 1990.*
16. S. Jeng and W. Tsai, Scale- and orientation-invariant generalized Hough transform—a new approach, *Pattern Recognit.* **24**, 1991, 1037–1051.
17. R. K. K. Yip, P. K. S. Tam, and D. N. K. Leung, Modification of Hough transform for object recognition using a 2-dimensional array, *Pattern Recognit.* **28**, 1995, 1733–1744.
18. P.-K. Ser and W.-C. Siu, A new generalized Hough transform for the detection of irregular objects, *J. Visual Comm. Image Represent.* **6**, 1995, 256–264.
19. A. S. Aguado, M. E. Montiel, and M. S. Nixon, Improving parameter space decomposition for the generalized Hough transform, in *Int. Con. on Image Process. ICIP'96, September 16–19, Lausanne, Switzerland, 1996*.
20. A. P. Pentland, Perceptual organization and the representation of natural form, *Artificial Intelligence* **26**, 1986, 292–331.
21. J. Princen, J. Illingworth, and J. Kittler, A formal definition of the Hough transform: Properties and relationships, *J. Math. Imaging Vision* **1**, 1992, 153–168.
22. S. R. Deans, Hough transform from the radon transform, *IEEE Trans. PAMI* **13**, 1981, 185–188.
23. G. H. Granlund, Fourier preprocessing for hand print character recognition, *IEEE Trans. Comput. (Short Notes)* **21**, 1972, 195–201.
24. C. T. Zahn and R. Z. Roskies, Fourier descriptors for plane closed curves, *IEEE Trans. Comput.* **21**, 1972, 269–281.
25. C. W. Richard and H. Hemami, Identification of three-dimensional objects using Fourier descriptors of the boundary curve, *IEEE Trans. SMC* **24**, 1974, 371–378.
26. E. Persoon and K.-S. Fu, Shape discrimination using Fourier descriptors, *IEEE Trans. SMC* **17**, 1977, 170–179.
27. T. P. Wallace and P. A. Wintz, An efficient three dimensional aircraft recognition algorithm using normalized Fourier descriptors, *Comput. Vision Graphics Image Process.* **13**, 1980, 99–126.
28. T. R. Crimmins, A complete set of Fourier descriptors for two-dimensional shapes, *IEEE Trans. SMC* **12**, 1982, 848–855.
29. F. P. Kuhl and C. R. Giardina, Elliptic Fourier features of a closed contour, *Comput. Vision Graphics Image Process.* **18**, 1982, 236–258.
30. D. Proffit, Normalization of discrete planar object, *Pattern Recognit.* **15**, 1982, 137–143.
31. L. T. Watson and L.G.S. Shapiro, Identification of space curves from two-dimensional perspective views, *IEEE Trans. PAMI* **4**, 1982, 469–475.
32. P. J. van Otterlo, *A Contour-Oriented Approach to Shape Analysis*, Prentice Hall, New York, 1991.
33. P. Rosin and S. Venkatesh, Extracting natural scales using Fourier descriptors, *Pattern Recognit.* **26**, 1993, 1383–1393.
34. L. H. Staib and J. S. Duncan, Boundary finding with parametrically deformable models, *IEEE Trans. PAMI* **14**, 1992, 1061–1075.
35. W. C. Y. Lam, K. S. Y. Yuen, and D. N. K. Leung, Fourier parameterization provides uniform bounded Hough space, in *Computer Analysis of Images and Patterns*, 5th. Int. Conf., CAIP'93, Budapest, Hungary, (Dimitri Chetverikov and Watter G. Kropatsch, Eds.), Lecture Notes in Computer Science, Vol. 719, Springer-Verlag, Berlin/New York, 1993.
36. W. E. L. Grimson and D. P. Huttenglocher, On the sensitivity of the Hough transform for object recognition, *IEEE Trans. PAMI* **12**, 1990, 255–274.
37. K. F. Lai and R. T. Chin, Deformable contours: modeling and extraction, *IEEE Trans. PAMI* **17**, 1995, 1084–1090.
38. H. Li and M. A. Lavin, Fast Hough transform: a hierarchical approach, *Comput. Vision Graphics Image Process.* **36**, 1986, 139–161.
39. J. Illingworth and J. Kittler, The adaptive Hough transform, *IEEE Trans. PAMI* **9**, 1987, 690–697.
40. J. Van Aken and M. Nova, Curve-drawing algorithms for faster display, *ACM Trans. Graphics* **4**, 1985, 147–169.
41. S. D. Shapiro, Use of the Hough transform for image data compression, *Pattern Recognit.* **12**, 1980, 333–337.
42. Y. C. Hecker, J. Illingworth, and J. Kittler, Detecting partially occluded ellipses using the Hough transform, *Image Vision Comput.* **7**, 1989, 31–37.
43. A. S. Aguado, M. E. Montiel, and M. S. Nixon, On using directional information for parameter space decomposition in ellipse detection, *Pattern Recognit.* **29**(3), 1996, 369–381.
44. D. H. Ballard and D. Sabbah, Viewer independent shape recognition, *IEEE Trans. PAMI* **5**, 1983, 653–660.